NAVAL POSTGRADUATE SCHOOL

MONTEREY, CALIFORNIA

MBA PROFESSIONAL REPORT

Benchmarks for Enhanced Network Performance: Hands-On Testing of Operating System Solutions to Identify the Optimal Application Server Platform for the Graduate School of Business and Public Policy

---

**By: Rex Burman, and
Anthony R. Coca**

September 2010

**Advisors: Albert Barreto,
Douglas Brinkley**

THIS PAGE INTENTIONALLY LEFT BLANK

REPORT DOCUMENTATION PAGE				*Form Approved OMB No. 0704-0188*	
Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instruction, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188) Washington DC 20503.					
1. AGENCY USE ONLY ***(Leave blank)***		**2. REPORT DATE** September 2010	**3. REPORT TYPE AND DATES COVERED** MBA Professional Report		
4. TITLE AND SUBTITLE Benchmarks for Enhanced Network Performance: Hands-On Testing of Operating System Solutions to Identify the Optimal Application Server Platform for the Graduate School of Business and Public Policy **6. AUTHOR(S)** Rex Burman and Anthony R. Coca.				**5. FUNDING NUMBERS**	
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Naval Postgraduate School Monterey, CA 93943-5000				**8. PERFORMING ORGANIZATION REPORT NUMBER**	
9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES) N/A				**10. SPONSORING/MONITORING AGENCY REPORT NUMBER**	
11. SUPPLEMENTARY NOTES The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government. IRB Protocol number _______N/A_________.					
12a. DISTRIBUTION / AVAILABILITY STATEMENT Approved for public release; distribution is unlimited				**12b. DISTRIBUTION CODE**	
13. ABSTRACT (maximum 200words) With the release of next generation operating systems, network managers face the prospect of upgrading their systems based on the assumption that "newer is better". The Graduate School of Business and Public Policy is in the process of upgrading their network application server and one of the most important decisions to be made is which Server Operating System to use. Based on hands-on benchmark tests and analysis we aim to assist the GSBPP by providing benchmark metrics and a recommendation of which Operating Systems will provide the best solution.					
14. SUBJECT TERMS Benchmark, Application Server, Performance Test, Microsoft Server 2008, Microsoft Server 2003, time-to-completion test, network.					**15. NUMBER OF PAGES** 107
					16. PRICE CODE
17. SECURITY CLASSIFICATION OF REPORT Unclassified	**18. SECURITY CLASSIFICATION OF THIS PAGE** Unclassified		**19. SECURITY CLASSIFICATION OF ABSTRACT** Unclassified		**20. LIMITATION OF ABSTRACT** UU

NSN 7540-01-280-5500

Standard Form 298 (Rev. 2-89)
Prescribed by ANSI Std. 239-18

THIS PAGE INTENTIONALLY LEFT BLANK

Approved for public release; distribution is unlimited

BENCHMARKS FOR ENHANCED NETWORK PERFORMANCE: HANDS-ON TESTING OF OPERATING SYSTEM SOLUTIONS TO IDENTIFY THE OPTIMAL APPLICATION SERVER PLATFORM FOR THE GRADUATE SCHOOL OF BUSINESS AND PUBLIC POLICY

Rex Burman, Lieutenant Commander, United States Navy
Anthony R. Coca, Lieutenant, United States Navy

Submitted in partial fulfillment of the requirements for the degree of

MASTER OF BUSINESS ADMINISTRATION

from the

NAVAL POSTGRADUATE SCHOOL
September 2010

Authors: ____________________________________
Rex Burman

____
Anthony R. Coca

Approved by: ____________________________________
Albert Barreto, Co-Advisor

____
Douglas Brinkley, Co-Advisor

____
William R. Gates, Dean
Graduate School of Business and Public Policy

THIS PAGE INTENTIONALLY LEFT BLANK

Benchmarks for Enhanced Network Performance: Hands-On Testing of Operating System Solutions to Identify the Optimal Application Server Platform for the Graduate School of Business and Public Policy

ABSTRACT

With the release of next generation operating systems, network managers face the prospect of upgrading their systems based on the assumption that “newer is better”. The Graduate School of Business and Public Policy is in the process of upgrading their network application server and one of the most important decisions to be made is which Server Operating System to use. Based on hands-on benchmark tests and analysis we aim to assist the GSBPP by providing benchmark metrics and a recommendation of which Operating Systems will provide the best solution.

THIS PAGE INTENTIONALLY LEFT BLANK

TABLE OF CONTENTS

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF FIGURES

LIST OF TABLES

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF ACRONYMS AND ABBREVIATIONS

1P	One Processor
8P	Eight Processors
3D	Three Dimensional
CAL	Client Access License
CPU	Central Processor Unit
DNS	Domain Name System
GPU	Graphics Processing Unit
GSBPP	Graduate School of Business and Public Policy
IT	Information Technology
ITAC	Information Technology Assistance Center
NPS	Naval Postgraduate School
LDAP	Lightweight Directory Access Protocol
OLE	Object Linked Embedded
OOB	Out of Box
PC	Personal Computer
PERFECT	PERFormance Evaluation by Cost-effective Transformations
SPEC	Standard Performance Evaluation Corporation

THIS PAGE INTENTIONALLY LEFT BLANK

ACKNOWLEDGMENTS

We would like to thank Professor Douglas Brinkley and Albert "Buddy" Barreto for their guidance and direction in the completion of our MBA Project. Their professional expertise was critical to our understanding of performance testing, network setup and data interpretation. The hand on time spent with us was especially helpful and we thank you for that. Also, we would like to thank Naval Postgraduate School intern Kayla Smith, for assistance and guidance with setting up Terminal Services and Domain Name Services within the Windows Server 2003 operating systems. We extend our gratitude to fellow NPS graduate students Justin Bummara, Mary Kessler, and Mark McClellan for their contributions on the OfficeBench tests, which were the catalyst for this project. Finally, we would like to thank Professor Elda Puma for answering questions and proposing solutions during our tests using the Stata application.

LCDR Rex Burman: I would like to thank God for giving me the strength over the past 21 months in completing this curriculum. To my daughter Jenelle, thank you for inspiring me to do my best in school. To my loving wife, Jenny, for her courageous sacrifice and strength during her pregnancy and bringing our son, Ridge, into this world. A special thank you goes out to Tony, for all your help and support. I look forward to working with you again soon.

LCDR Anthony R. Coca: I would like to offer my most heartfelt gratitude to my loving wife, Valerie, and my two children, Tristen, and Lauren. Your understanding and support was critical to the successful completion of this project. Your sacrifices and our time spent apart are too many, yet you carry on with grace and dignity. I would also like to thank my project partner Rex for his professionalism and assistance on this project. See you in Pennsylvania.

THIS PAGE INTENTIONALLY LEFT BLANK

I. INTRODUCTION

A. PURPOSE

The goal of this project is to answer the following question: which server operating system is the best fit for a new and robust application server for the Graduate School of Business and Public Policy (GSBPP). To do this we will use a series of test scripts and benchmark results to measure and analyze the speed (time-to-completion) of several common business work-tasks across four different operating systems. Our ultimate goal is to share our test results and provide a metric that assists GSBPP with its selection of an application server architecture operating system.

B. BACKGROUND

The Naval Postgraduate School's Graduate School of Business and Public Policy (GSBPP) endeavors to provide its students and faculty with the most up to date and best performing technology it can. GSBPP is in the process of updating its computer assets with the anticipation of evolving hardware and software demands. Within the scope of this upgrade, the school seeks to provide its members the ability to conduct resource heavy modeling and robust computing capability. To facilitate the best use of computing resources and funding the GSBPP Information Technology Directorate requires benchmarking analysis in order to procure the best-fit operating system for its future application server.

GSBPP has approximately 250 resident students, which are enrolled in various Master's degree programs at any point in time (Graduate School of Business and Public Policy, Naval Postgraduate School, 2010). With this number of students and the depth of studies/research that occurs, there exists a need for an application server that can provide the necessary resources to the staff and students to perform the heavy modeling, statistical analysis, and database administration performed.

C. GUIDANCE

Our guidance was to assess the relative performance of varying flavors of Windows Server operating systems. To do this we conducted a literature review to discover what experts in this area had accomplished during similar studies, gain insight as to the history / evolution of the systems we were testing and develop a workflow that would allow us to achieve our goals.

Additional guidance from the GSBPP Information Technology Director dictated that each operating system should be stressed to record the time required to access, retrieve, and create content in Microsoft Office 2007, Microsoft Office 2003, and Stata 11 I/C statistical analysis software. Methodical and scripted application of each benchmarking tool was essential to gathering quality results. Time-to-completion speed was the primary metric gathered; time-to-completion can be defined as how quickly a system completes a test script or task. The benchmark results in this paper do not account for the functionality, reliability, or stability of the operating systems, simply the speed in which the benchmark was completed.

Two secondary goals were also set. The first being to perform a benchmark test on two different client operating systems (Windows Professional 7 and Windows Professional XP) and stress them across an ad hoc network in tandem with the server operating systems in our study. The goal is to compare time-to-completion results when pairing each combination of the test subject operating systems across a client – server network.

The second was to research and make clear the process of procuring and licensing a Windows Server operating system.

A key component of this project was to ensure each test was done fairly and without bias. In support of this requirement we ensured that the same hardware architecture was used during each operating system test. In addition, each application server instance would have the same application load and be up to date with all Microsoft updates and patches. Through this approach, we were able to establish a workflow that adhered to the restrictions and requirements set forth by the GSBPP IT Director.

II. LITERATURE REVIEW

The purpose of this section is to establish the knowledge we gained on the topic benchmarking through studying the work of experts in the field of computer benchmarking. It also provides the ability to bring the reader up to date on current and past research in the field and provides alternative / contrasting views on the topic. This section of our MBA project provides us with the opportunity to:

- Research and report on what has (or has not been) found in the area of benchmarking in general and in the operating systems we are researching.
- Gain knowledge of the methodology and reports provided by the benchmark tools
- Identify relationships between concepts.
- Define key concepts.
- Identify data sources that other researchers have used.

To begin we review literature that establishes basic concepts on the subject of benchmarking computer systems and builds a foundation of general benchmark limitations and benefits.

A. THE SCIENCE OF COMPUTER BENCHMARKING

The author, Roger Hockney, begins by providing basic benchmarking context and ideas. This book was originally written as literature to support tutorial classes taught by the author at an event called Supercomuting 94 in Washington D.C. This event would give rise to an influential group in the field of computer benchmarking known as PERFECT. PERFECT would go on to build many industry standard benchmark tools (Hockney, 1996, p. 2). The bulk of the author's benchmark experience comes from testing parallel computing systems, which are different than the systems that will be tested in the project. However, the concepts are universal and are pertinent to our project.

We begin by considering the benefits of computer benchmarking. It is important to note that a benchmark test never provides all the information needed, however, when done correctly a benchmark (or performance test) can provide detailed information in critical areas if you know what you are looking for and know how to ask for it. In doing

so, benchmark tests present important data about computer systems and they should not be discounted (Hockney, 1996, p. 10). This data can then serve to guide with the acquisition of a system or assist a developer with improving the performance of a system.

The author then outlines a benchmarking approach based on testing a small (or a limited) number of parameters. By only testing a small number of parameters, accuracy is increased and results can be compared against other systems much more easily with a greater depth of understanding. This is known as "benchmarking for knowledge" (Hockney, 1996, p. 11). It is also the same approach we will use in our project since we will concentrate primarily (but not always be limited to) time-to-completion of tasks to gain knowledge of how quickly an operating system completes commonly performed business tasks with popular software suites.

1. Benchmark Limitations

The author summarizes the following limitations:

- Benchmarks do not answer questions that are not asked
- Specific application benchmarks cannot tell you (or provide) about the performance of another application.
- To understand the results of a benchmark requires that you know the background purpose (Hockney, 1996, p. 12).

2. Levels of Benchmark Usefulness

The book argues then that most useful type of benchmark test is a low-level benchmark. Low-level benchmarks measure the basic capabilities of computer hardware as impacted by the test being performed on it. They also measure the fundamental results of the computer architecture. These low level tests are also known as kernel benchmarks (Hockney, 1996, p. 12).

Down a notch in levels of usefulness is the application benchmark. This is because it is difficult to draw generalized conclusion from such a test if you do not understand the background of the benchmark; this may lead to a case of not understanding or misinterpretation of results (Hockney, 1996, p. 13).

We use both types of these benchmarks in our study.

B. PURPOSE-BASED BENCHMARKS

Purpose-Based Benchmarks was written in order to define a performance test approach to "measures the ability of a computing system to reach a goal of human interest" (Gustafson, 2004, p. 1). Although this approach is somewhat proprietary to the author's studies, the knowledge and ideas gained from this article foster a deeper understanding of benchmarking and where the field may be headed.

1. Benchmark Design Goals

Benchmarks are performance tests of computer systems, which serve the following purposes:

- Assist users in estimating the performance of a system on their workload prior to purchase
- Assist system designers in optimizing their designs before finalizing their choices

The author argues that these purposes imply that benchmarks should be low-cost or quick when compared to running a full customer workload or building a system and measuring performance on the actual system (Gustafson, 2004, p. 1). By not employing quicker or lower cost test, the authors argue that impactful results may emerge too late or be too expensive to employ.

This thought strengthens the validity of this study, since it is in line with our purpose, which is to provide a relevant recommendation of which server operating system is the best-fit system for the GSBPP based on quick and low cost benchmark tests and analysis.

2. Speed as a Measure of Performance

This is an important concept to understand since it is the metric we will most often measure.

The most common metric of a computer benchmarks is "speed." However, speed is not a well-defined measure and often lacks properties that are measurable in physics. It is difficult to apply conventional measures like miles per hour or temperature to a computer system benchmark. Computer speed is the product of workload and time;

however "workload" can be an arbitrary term. The standard workaround for this is to attempt to define a fixed task as the work and use the reciprocal of the execution time as the speed (Gustafson, 2004, p. 1). This is the approach used by the majority of the tools used in this study.

3. Benchmark Tools and Gaps

As a benchmark becomes widely used there is an unintended consequence that commonly occurs; the area studied becomes overvalued. This leads to the possibility that areas that are not tested may be neglected, despite their importance.

As a result, top benchmark designers create large test suites, with the idea of covering every aspect. The drawback of this is that the cost of producing and running the benchmark becomes costly and still may not cover every aspect. The author points to SPEC and PERFECT Club as examples of this (Gustafson, 2004, p. 3).

Our team explored the possibility of using products developed by both of these industry-leading developers. After consideration, we decided that the cost was too high for the measures and budget constraints we experienced. Although, we admit that the level of results from SPEC benchmark tools would have produced a deeper level of validity than the tests we adopted; it was agreed that the tools we used were the right fit for this projects goals and constraints.

There are also two traps that we must be aware of with regard to benchmarking a system. The first being since we are only testing a small subset of our machine and associated operating systems we might assume that the untested properties are similar in performance. The second is that as a measurer we might provide a recommendation to the GSBPP without disclosing that our tests only cover a subset of the machine properties and that the performance of everything that is not tested is unimportant (Gustafson, 2004, p. 2). This is certainly not the case as reliability, functionality, and hardware / software symbioses are other areas that the GSBPP may want to consider as well. We reiterate once again that our primary metric for research is time-to-completion of common business application tasks.

4. Benchmarking Principles

The following information should be provided, along with any benchmark report:

- The date the test was made
- Who ran the test and how they may be contacted
- The precise description of what the test conditions were, sufficient that someone else could reproduce the results within statistical errors
- The software that was used, and an explanation for any modifications made to what is generally available as the definition of the benchmark
- An accounting of cost, including the published price of the system and any software that was used in the run.
- An accounting, even if approximate, of the amount of time spent porting the benchmark to the target system.
- Admission of any financial connections between vendor and the reporter; was the system a gift? Do they work directly for the vendor or for a contractor of the vendor?
- The range of results observed for the test, not just the most flattering results. The reporters should reveal the statistical distribution, even if there are very few data points (Gustafson, 2004, p. 3).

In keeping with these principles, our team has given all of this information within this document. Areas that have not been covered elsewhere are addressed here, starting with test dates:

- November 2009 – OfficeBench
- July 2010 – PerformanceTest
- July 2010 – NovaBench
- August 2010 – Startup/Shutdown Times
- August 2010 – PERFMON Tests

All benchmark test tools used in this project had zero monetary cost to the GSBPP. That is to say that all were developed as shareware with fully functioning trial periods or were free and openly available.

The costs of the Operating Systems and Microsoft Office Suites changes depending on vendor but the general range of these products is in the \$250–\$600 price range. Licensing costs will be covered elsewhere.

A combined total amount of time spent conducting these benchmarks is 42 hours. This time estimate includes setup of the server machine, the ad hoc network and research. It does not include the time allotted to creating this report.

Our team is comprised of graduate students at the Naval Postgraduate School without financial connections between the team and the vendors of the products used.

C. WINDOWS SERVER 2003

Windows Server 2003 is first of our candidate server operating systems to be benchmarked in this project. It is important to review the context and history of this operating system, since it is sheds light on previous benchmark tests and the functionality of Windows Server 2003, which will aid in a producing a valid recommendation of an operating system that will best serve the needs of the GSBPP.

Released on April 24, 2003, Windows Server 2003 was designed to be the leader in trustworthy computing. According to a 2002 memo from Microsoft Chairman and Chief Software Architect, he challenged his company to improve the experiences of Microsoft customers in regards to a concept that Microsoft has come to call the "four pillars" of trustworthy computing; the Trustworthy Computing Initiative four pillars are security, privacy, reliability, and business integrity (Montehermosa, 2004).

Five editions of Windows Server 2003 were introduced (Standard, Enterprise, Datacenter, Web, and Small Business 2003) with Storage Server and Compute Cluster being released at a later date.

Table 1 lists the minimum requirement for Windows Server 2003.

Windows Server 2003 System Requirements				
Requirement	**Standard Edition**	**Enterprise Edition**	**Datacenter Edition**	**Web Edition**
Minimum CPU Speed	133 MHz	• 133 MHz for x86-based computers • 733 MHz for Itanium-based computers*	• 400 MHz for x86-based computers • 733 MHz for Itanium-based computers*	133 MHz
Recommended CPU Speed	550 MHz	733 MHz	733 MHz	550 MHz
Minimum RAM	128 MB	128 MB	512 MB	128 MB
Recommended Minimum RAM	256 MB	256 MB	1 GB	256 MB
Maximum RAM	4 GB	• 32 GB for x86-based computers • 64 GB for Itanium-based computers*	• 64 GB for x86-based computers • 512 GB for Itanium-based computers*	2 GB
Multiprocessor Support **	Up to 4	• Up to 8	• Minimum 8-way capable machine required • Maximum 64	Up to 2
Disk Space for Setup	1.5 GB	• 1.5 GB for x86-based computers • 2.0 GB for Itanium-based computers*	• 1.5 GB for x86-based computers • 2.0 GB for Itanium-based computers*	1.5 GB

* **Important:** The 64-bit versions of Windows Server 2003, Enterprise Edition and Windows Server 2003, Datacenter Edition are only compatible with 64-bit Intel Itanium-based systems. They cannot be successfully installed on 32-bit systems.

** Windows Server 2003 may not use multiple processors with some Intel Pentium Pro or Pentium II Processors.

Table 1. Windows Server 2003 Requirements (From Microsoft Corporation, 2006)

There are several key performance components that set Windows Server 2003 apart at the time of its release. Advances in hardware synergy, a redesigned intuitive architecture and enhanced common services were critical updates to the server operating system, which are still needed today. These improvements were valuable to IT managers implementing cost-saving measures, such as consolidation, and helped developers relieve performance bottlenecks in certain applications (Alliegro, 2003). Such is the case with Microsoft Office. Microsoft Corporation had commissioned VeriTest to perform a series of benchmarks comparing Windows Server 2003 Enterprise Edition against Windows Server 2000 Advance Server and Windows NT 4.0. The next three graphs are results taken from the VeriTest study. Figure 1 shows the results of a file server’s performance using NetBench Benchmark.

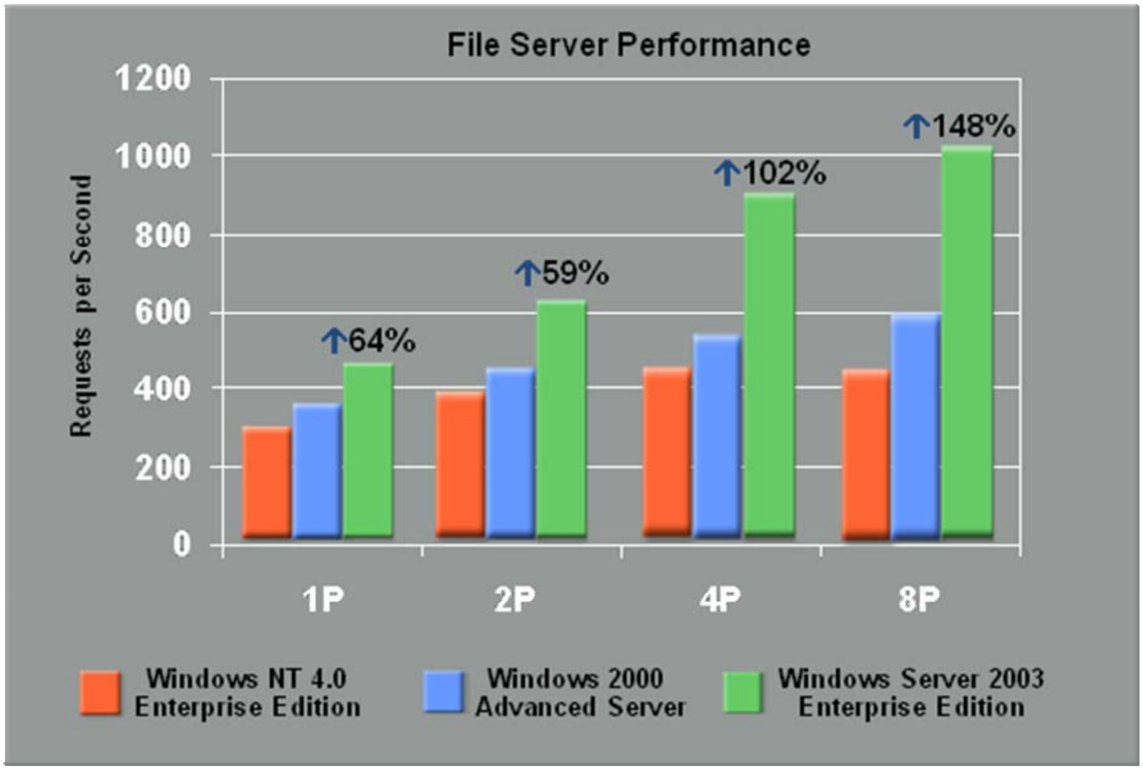

Figure 1. File Server Performance Windows Server 2003 (From Alliegro, 2010)

The graph measures how well a file server handles file in and out requests through network file operations from 32-bit Windows clients. Using a single processor, the benchmark shows how Windows Server 2003 performed 64% faster than its predecessor, Windows NT 4.0 and up to 148% faster using an eight-processor system. Additional performance benchmarks included Web Server Performance and Active Directory Performance. The Web Server Performance benchmark measured the performance of Static Mix, Dynamic (ISAPI and CGI) Mix, and E-Commerce Mix. All benchmarks provided the results of how Windows Server 2003 Enterprise Edition has outperformed Windows NT 4.0 Enterprise Edition and Windows Server 2003 Advanced Server.

For the Active Directory Performance (Messaging Mix and Address Mix) benchmark, VeriTest compared Windows Server 2003 against Windows Server 2000 Advance Server. The Messaging Mix benchmark (Figure 2) simulated and e-mail server's use of a directory based on LDAP protocol and a one million-user database with more than ten organizational units.

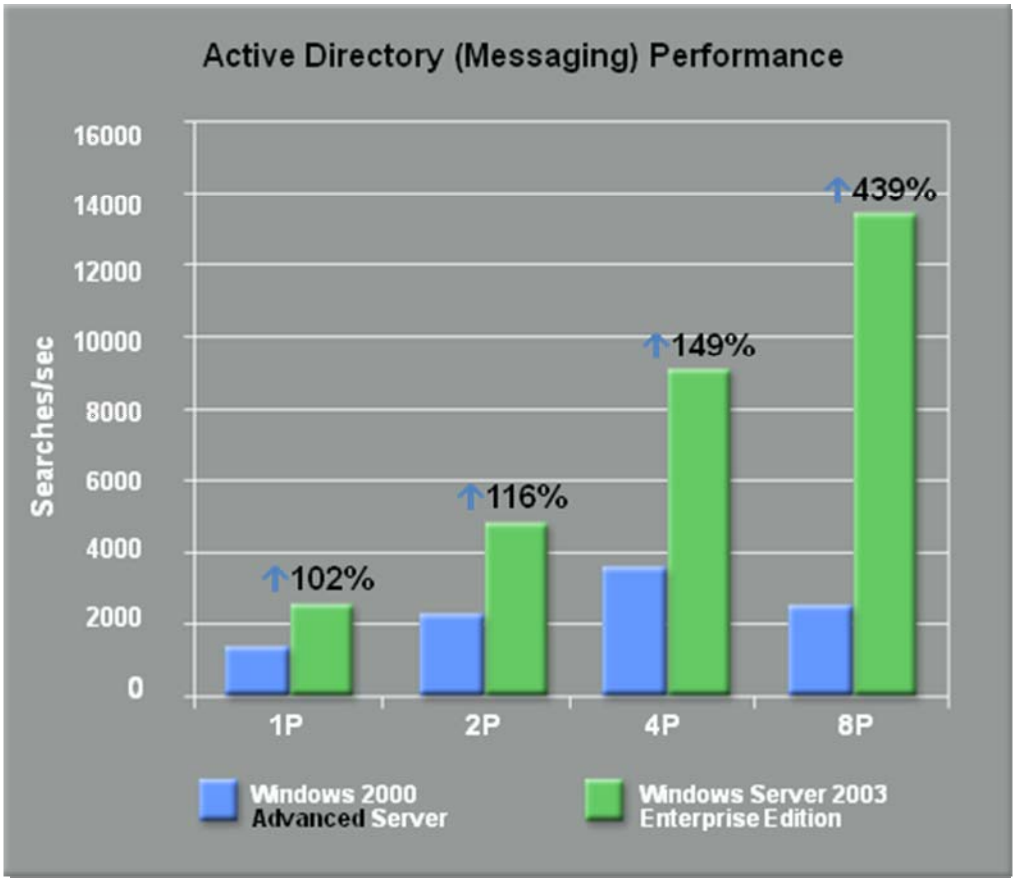

Figure 2. Active Directory (Messaging) Performance Windows Server 2003 (From Alliegro, 2010)

The results show that Windows Server 2003 is 102% faster in a 1P system than Windows 2000 Advance Server and 439% faster with an 8P system. Similar to the Messaging Mix, the Address Mix (Figure 3) simulated users looking up names in address book and expanding group for e-mail based on LDAP and a one million-user database with more than ten organizational units.

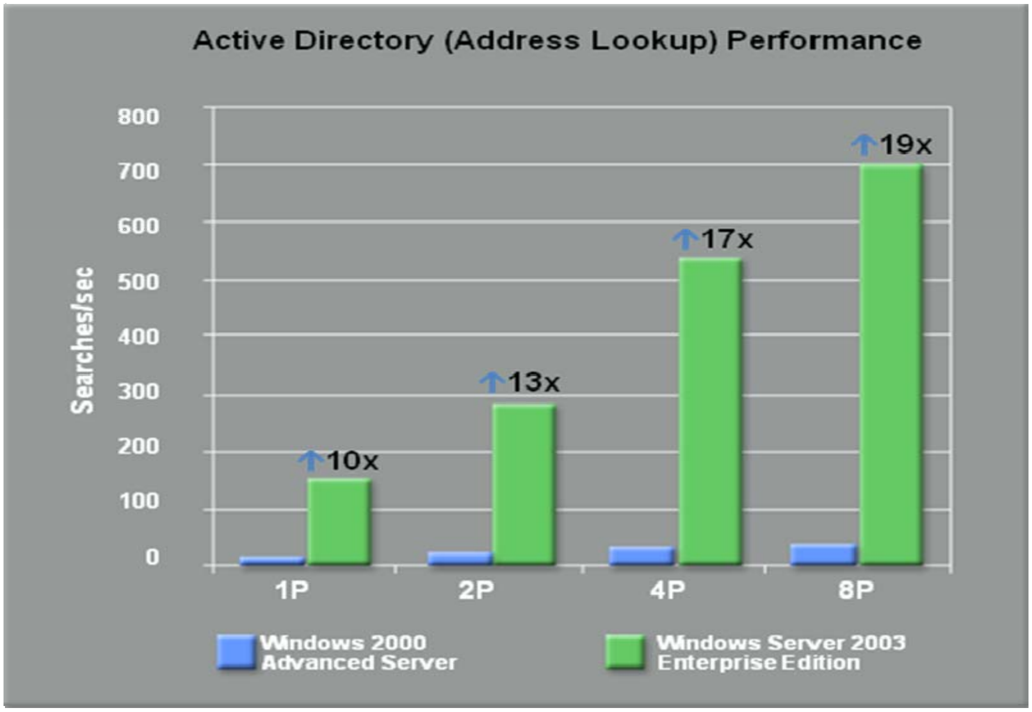

Figure 3. Active Directory (Address Lookup) Performance Windows Server 2003 (From Alliegro, 2010)

Again, the benchmark results show that Windows Server 2003 out performed Windows 2000 Advanced Server, up to ten times faster using a 1P system and nineteen times faster using an 8P system.

The hardware used for all benchmarks include the following:

- HP ProLiant DL760
- Intel Pentium III Xeon 900 MHz processors
- 4 GB RAM
- Intel PRO/1000 MF Server Adapters

While these benchmark tests do not exactly compare the operating systems that we will test, nor do they use the tools we utilize; it is important to show these tests since they act as a template for many of the tests we will use. This further validates the methods we will use since IT professionals at Microsoft Corporation use similar methodology and technique.

D. WINDOWS SERVER 2008

Succeeding Windows Server 2003 is Windows Server 2008, which was released on February 27, 2008. A primary purpose of Windows Server 2008 is to ensure that the operating system can be optimized for use in any networking scenario (Stanek, 2008). Like its predecessor, multiple editions of the product were released and are available in 32- or 64-bit versions, all intended to be replacements for the Windows Server 2003 versions. The four main product editions are:

- Windows Server 2008 Standard
- Windows Server 2008 Enterprise
- Windows Server 2008 Data Center
- Windows Web Server 2008

Windows Server 2008 R2 was launched on 10/22/2009. Version upgrades include new functionality for Active Directory, new Virtualization and Management features and support for up to 256 logical processors. Table 2 shows the systems requirements for Windows Server 2008 R2.

Component	Requirement
Processor	Minimum: 1.4 GHz (x64 processor) Note: An Intel Itanium 2 processor is required for Windows Server 2008 R2 for Itanium-Based Systems
Memory	Minimum: 512 MB RAM Maximum: 8 GB (Foundation) or 32 GB (Standard) or 2 TB (Enterprise, Datacenter, and Itanium-Based Systems)
Disk Space Requirements	Minimum: 32 GB or greater Note: Computers with more than 16 GB of RAM will require more disk space for paging, hibernation, and dump files
Display	Super VGA (800 × 600) or higher resolution monitor
Other	DVD Drive, Keyboard and Microsoft Mouse (or compatible pointing device), Internet access (fees may apply)

Table 2. System Requirements–Windows Server 2008 [1] (From Microsoft Corporation, 2010b)

[1] Actual requirements will vary based on your system configuration, and the applications and features you choose to install. Processor performance is dependent upon not only the clock frequency of the processor, but the number of cores and the size of the processor cache. Disk space requirements for the system partition are approximate. Additional available hard disk space may be required if you are installing over a network Microsoft Corporation, Windows Server 2008 R2 System Requirements, 2010, 2010 17-August.

The product also incorporated key architectural improvements in the Kernel architecture, support architecture, and in the boot environment—which makes Windows Server 2008 the first truly hardware independent version of the Windows Server Operating System (Stanek, 2008).

Another key feature of Windows Server 2008 is the server virtualization with Hyper-V technology. Hyper-V is built on 64-bit hypervisor technology, allowing increase in workloads to be managed effectively, such as workloads that involve 32- and 64-bit processors. With Hyper-V, underutilized physical servers can be consolidated into virtual servers running on a single physical server, making a few optimized servers do the work of many underutilized servers (Microsoft Corporation, 2010a). For one particular company, Kroll Factual Data, they were able to run one thousand virtual machines on only 275 physical servers. Not only were there production benefits, but there were costs benefits as well. Says Chris Steffan, Information Security and Compliance Manager of Kroll Factual Data, “Consider how much power we save every day by running 20 virtual servers on one machine, versus 20 stand-alone servers,” Steffen notes. “If we took every virtual machine and ran it on a stand-alone machine like we did before, our power costs would at least double—not to mention the other costs accrued in additional equipment, real estate, networking, and support”(Microsoft Corporation, 2010a).

With energy saving measures on the minds of many consumers, Microsoft Corporation touts the power savings benefits of Windows Server 2008. A study conducted by Stanford University found that the amount of electricity used by servers and auxiliary equipment worldwide more than doubled between 2000 and 2005, to more than 1.2 billion kilowatt-hours (kWh) yearly—a figure that represents 0.8 percent of the estimated world electricity sales (Microsoft Corporation, 2010a). In order to meet the cost savings demands and environmental concerns of many governments, businesses, and individual consumers, Windows Server 2008 was designed to be more energy efficient and have various power-saving features.

1. Out-of-the-Box Power Savings (OOB)

Microsoft tested the power consumption of both Windows Server 2003 and Windows Server 2008 when immediately installed with OOB configurations. A single server with two dual-core processors and 4GB of RAM was used. Identical file operations tests were conducted with escalating load levels up to the maximum load level the system could accommodate, and power consumption was monitored (Microsoft Corporation, 2008a). The OOB test results revealed that Windows Server 2008 was able to save power up to ten percent at comparable levels of throughput against Windows Server 2003 (in Microsoft's controlled environment). Figure 4 illustrates how each server consumed power and how Windows Server 2003 achieved only 80% of maximum throughput compared to Windows Server 2008.

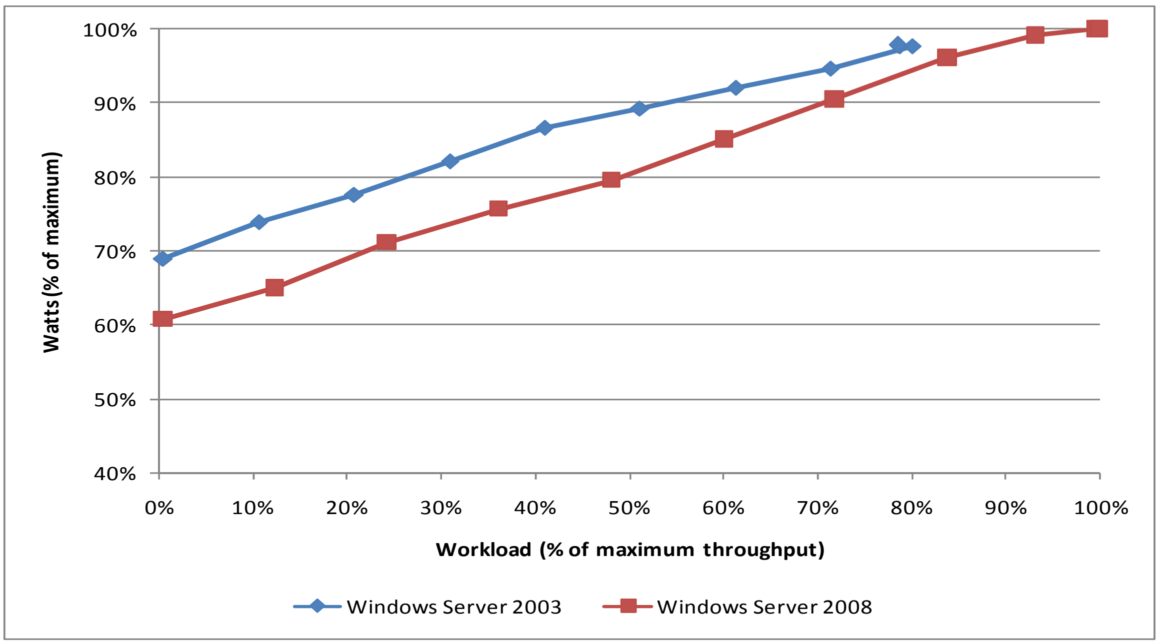

Figure 4. Workload (Windows Server 2003 vs. Windows Server 2008) (From Microsoft Corporation, 2008a)

2. Idle vs. Active Power Consumption

Another test that was conducted was the power consumption between Windows Server 2003 and Windows Server 2008 during idle and active periods. An enterprise-class server with 4 quad-core processors, 16 gigabytes of RAM, and a 288GB RAID5

hard disk array was configured with Windows Server 2003 running Internet Information Services 6 (Microsoft Corporation, 2008c). Using the same physical hardware, Windows Server 2008 running with Internet Information Services 7 replaced Windows Server 2003 and the test resumed to monitor its energy usage. Figure 5 illustrates the results after four tests were performed on the hardware. Using the default OOB settings and running on the same hardware, Windows Server 2008 clearly used less power than Windows Server 2003. Savings ranged between 2.3 percent at idle and 6.8 percent when the servers were actively servicing requests (Microsoft Corporation, 2008c).

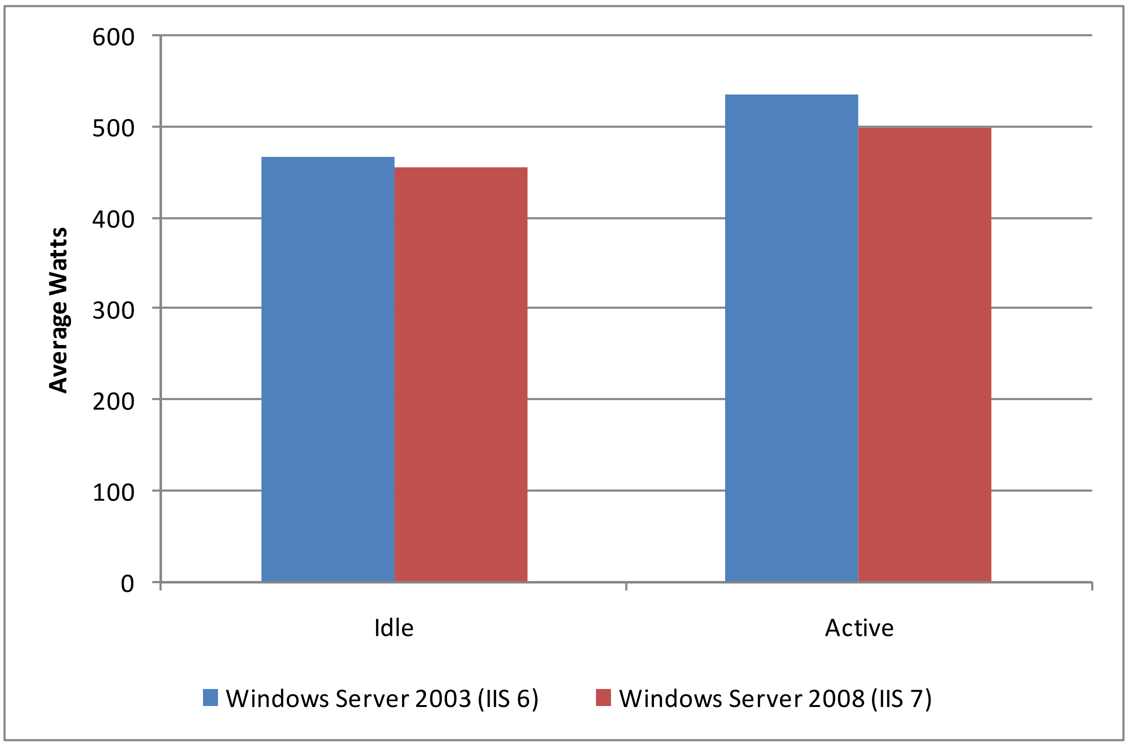

Figure 5. Energy Consumption Test: Windows Server 2003 vs. Windows Server 2008 (From Microsoft Corporation, 2008c)

E. TOLLY GROUP STUDY

To prove its effectiveness, the Microsoft Corporation commissioned another independent study to compare Windows Server 2008 against Windows Server 2003. The Tolly Group conducted various performance benchmarks and analysis comparing Windows Server 2008 running with Windows Vista clients against Windows Server 2003

running with Windows XP clients. In many test scenarios, the tandem of Windows Vista and Windows Server 2008 delivered the greatest performance gains of any client/server operating system combination, and yielded the most impressive time-to-completion of tasks performed (The Tolly Group, 2008).

Figure 6 records the results of the average throughput and time to completely open a 20MB Microsoft Office Excel file across a LAN.

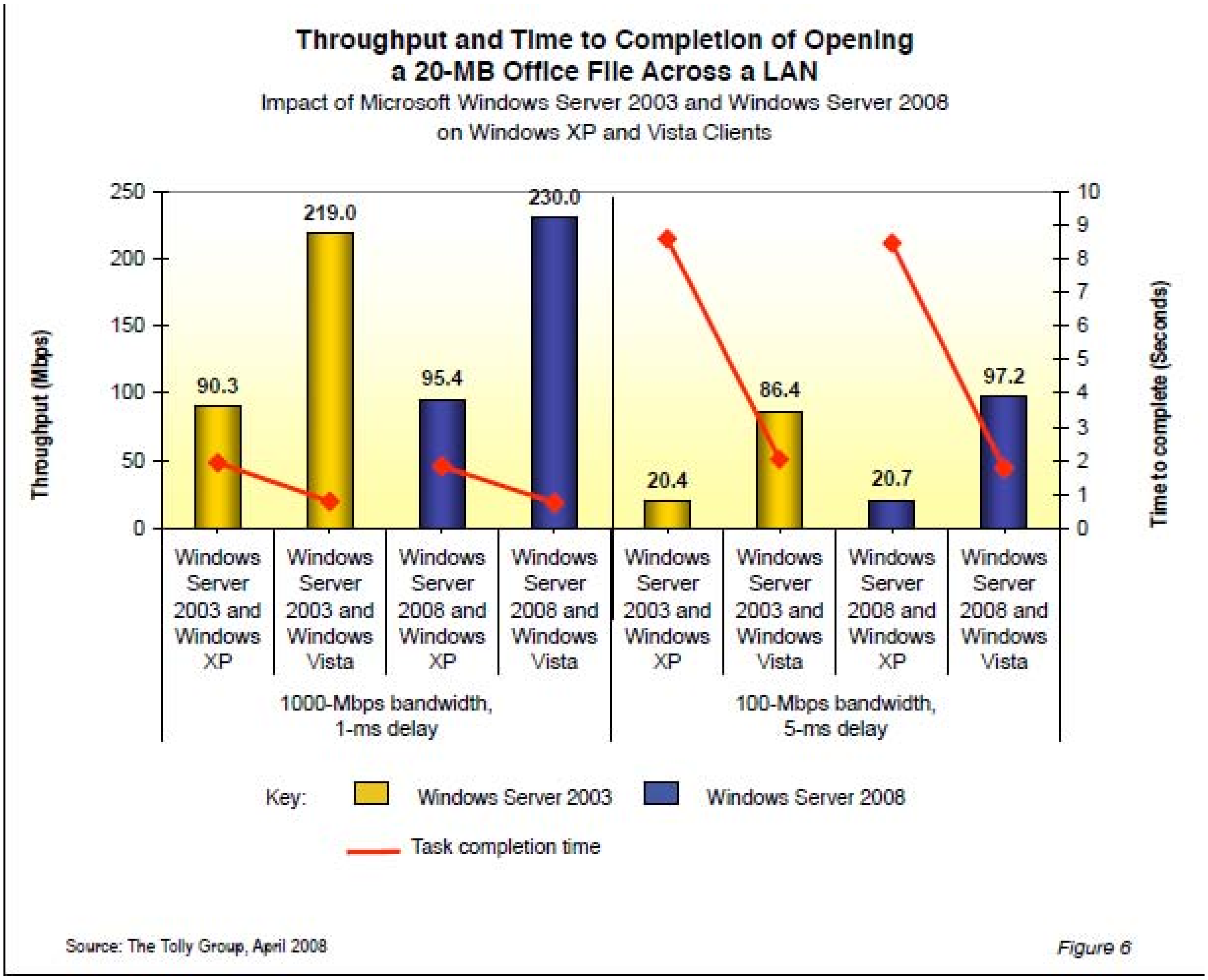

Figure 6. Throughput and Time to Completion of Opening a 20MB File (From Tolly Group, 2008)

- The file open task completed in 1.93 seconds between a Windows Sever 2003 R2 server and Windows XP; average data throughput was 90.35 Mbs (The Tolly Group, 2008).

- The file open task completed in 1.83 seconds, and throughput was 95.39 Mbs between a Windows XP client and a Windows Sever 2008 server (The Tolly Group, 2008).

Throughout the Tolly Group's study, Windows Server 2008 outperformed Windows Server 2003 when running on Windows XP. There were significant performance enhancements when running Windows Vista. Their conclusion points to the fact that organizations should seriously examine upgrading Windows server operating systems to Windows Server 2008 to extract the maximum performance out of WAN and LAN connections (The Tolly Group, 2008).

F. WINDOWS XP

A successor to Windows 2000 operating system, Windows XP was first released in August 2001 as a means to enhance the Windows experience for personal computer users. Windows XP was designed to integrate the strengths of Windows 2000, such as security, manageability, and reliability, with the best features of Windows 98 and Windows Millennium (Goktepe, 2002). Windows XP operating system was first released on October 25, 2001 with multiple editions and service packs being released in the years following. The XP name is short for "experience," symbolizing the rich and extended user experiences Windows and Office can offer by embracing Web services that span a broad range of devices (Microsoft Corporation, 2001). Three Service Packs (SP1, SP2, and SP3) were released to resolve any problems and add features to the operating system. The minimum hardware requirements for Windows XP Home and Professional Editions are:

The minimum hardware requirements for Windows XP Professional include:

- Pentium 233-megahertz (MHz) processor or faster (300 MHz is recommended)
- At least 64 megabytes (MB) of RAM (128 MB is recommended)
- At least 1.5 gigabytes (GB) of available space on the hard disk
- CD-ROM or DVD-ROM drive
- Keyboard and a Microsoft Mouse or some other compatible pointing device

- Video adapter and monitor with Super VGA (800 x 600) or higher resolution
- Sound card
- Speakers or
- Headphones (Microsoft Corporation, 2007b).

1. Benchmarking of Windows XP

Microsoft Corporation used the following commercial benchmarking tools to assess the performance and development of Windows XP:

- eTesting Labs' Business Winstone 2001 and Content Creation Winstone 2001
- BAPCo's Webmark 2001 and SysMark 2001
- PC World's PCWorldBench
- MadOnion's 3DMark 2000
- ETestingLabs' 3D WinBench 2000
- Microsoft-developed benchmarks (Fortin, 2010).

2. Benchmarked Applications

While conducting the benchmarks for Windows XP, Microsoft used applications that covered a broad spectrum of user interests and needs. They provided a broad scope for determining application requirements and behaviors, and were essential for understanding how Windows XP would really behave during daily use (Fortin, 2010).

Microsoft used the following applications for their benchmark:

a. Web Browsers

- Netscape Navigator
- Microsoft Internet Explorer

b. Office Productivity

- Microsoft Word, Microsoft Excel, Microsoft Access, Microsoft PowerPoint, Microsoft FrontPage, Microsoft Outlook, and Microsoft Project
- Lotus Notes
- Quicken (Intuit)

c. Multimedia

- Adobe Photoshop and Premiere
- Corel Photopaint
- Sonic Foundry's Sound Forge
- Macromedia's Dreamweaver

d. Document and Multimedia Content

- Microsoft Windows Media™ Player and Microsoft NetMeeting®
- Adobe Acrobat
- Macromedia's Flashplayer
- Cycore's Cult3D
- Apple's QuickTime player
- Dragon System's Naturally Speaking
- A variety of games were also used (Fortin, 2010).

The results from the benchmark test of Windows XP showed significant improvement from its predecessor, Windows 2000. Windows XP provides excellent overall performance, this includes dramatically faster boot and resume times, along with highly responsive applications (Fortin, 2010).

3. Windows XP Service Pack 2 (SP2)

Windows XP SP2 is required on computers that have multiple CPUs that support ACPI processor performance states (Microsoft Corporation, 2001). This requirement includes computers that support the following items:

- Multiple physical sockets
- Multiple-core designs
- Multiple logical threads, such as Intel hyper-threading technology (Microsoft Corporation, 2001).

Because Windows XP was not originally designed to support performance states on multiprocessor configurations, changes are required to correctly realize this support on multiprocessor systems (Microsoft Corporation, 2001). Required changes to the kernel power manager are part of the Windows XP Service Pack 2. It is because of these changes that allow for Windows XP to correctly function on multiprocessor systems with processor performance states.

G. WINDOWS VISTA

The successor to Windows XP, Windows Vista, was released to the world on 01/30/2007. The visual style of the graphical user interface was one of the more noticeable changes in the operating system. The new graphical user interface was designed to be more aesthetically pleasing and provide a livelier Windows experience for the user. Although some features of Windows Vista may require additional or advanced hardware, systems requirements for the operating system are:

- 800 megahertz (MHz) processor and 512 MB of system memory
- 20 GB hard drive with at least 15 GB of available space
- Support for Super VGA graphics
- CD-ROM drive (Microsoft Corporation, 2010c).

Expected to be more advanced and secure than Windows XP, many users claim that Windows Vista performed worse than its predecessor. Independent testing and benchmarking of Windows Vista were conducted to compare their functionality.

1. Benchmarking of Windows Vista

In 2007, Pfeiffer Consulting conducted a benchmarking project to test the results of Microsoft's Windows Vista compared against Windows XP and also Apple's Mac OS X operating systems. Their main focus was on User Interface Friction. Pfeiffer Consulting uses the term User Interface Friction to define the difference in fluidity and productivity that can be observed when running the same program or procedure on different computer systems, or when trying to achieve the goal on tow similar digital devices (Pfeiffer Consulting, 2007). The following hardware was used to perform their benchmark:

- Dual 2.8GHz Dell Dimension workstation
- 3.2 GHz Dell XPS workstation
- Dual 2GHz iMac workstation
- 3GHz Mac Pro workstation (Pfeiffer Consulting, 2007).

Three distinct aspects of User Interface Friction were conducted.

a. Menu Latency Measures

Menu latency was measured by accessing menus and submenus according to varying usage patterns (a single sub-menu, two specific sub-menus situated in different menus, 3 different submenus in three distinct menus). Each operation was executed several times in succession, and each set of operations was clocked several times (Pfeiffer Consulting, 2007).

b. Desktop Operations

Using the same methodology, three different, frequently performed operations were executed repeatedly: creation of a new folder on the desktop, deleting an item using right-click, and opening several folder-windows in succession (Pfeiffer Consulting, 2007).

c. Mouse Precision

Essential for any task where precise positioning of the cursor is essential; the lack of precision can affect daily tasks such as menu selection, clicking on icons in an application or targeting hyper-links on a web page (Pfeiffer Consulting, 2007).

2. Benchmarking Results

Although Windows Vista scored points for its new look and design, it did not perform as well as its predecessor (and neither performed better than the Mac OS X). On average, the following variations were noted: Windows Vista was 20% slower than Windows XP in menu latency benchmarks; Windows Vista was the slowest in desktop operations scoring an average of 2.73 seconds per operation, compared to 2.34 for Windows XP, and 1.50 seconds for Mac OS X; and Windows Vista scored more mouse precision errors than Windows XP and Mac OS X (Pfeiffer Consulting, 2007). Figures 7 through 9 illustrate the benchmarks conducted between Windows Vista and Windows XP.

The results of these benchmarks are the reason we did not choose Windows Vista as a potential client operating system in our tests.

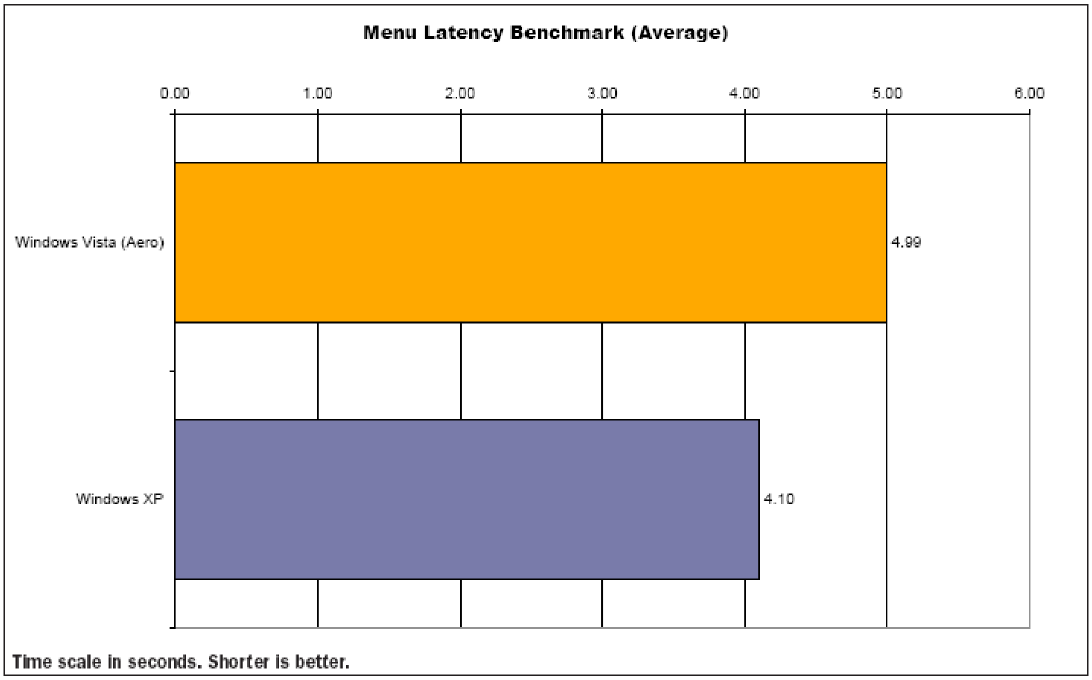

Figure 7. Menu Latency (From Pfeiffer Consulting, 2007)

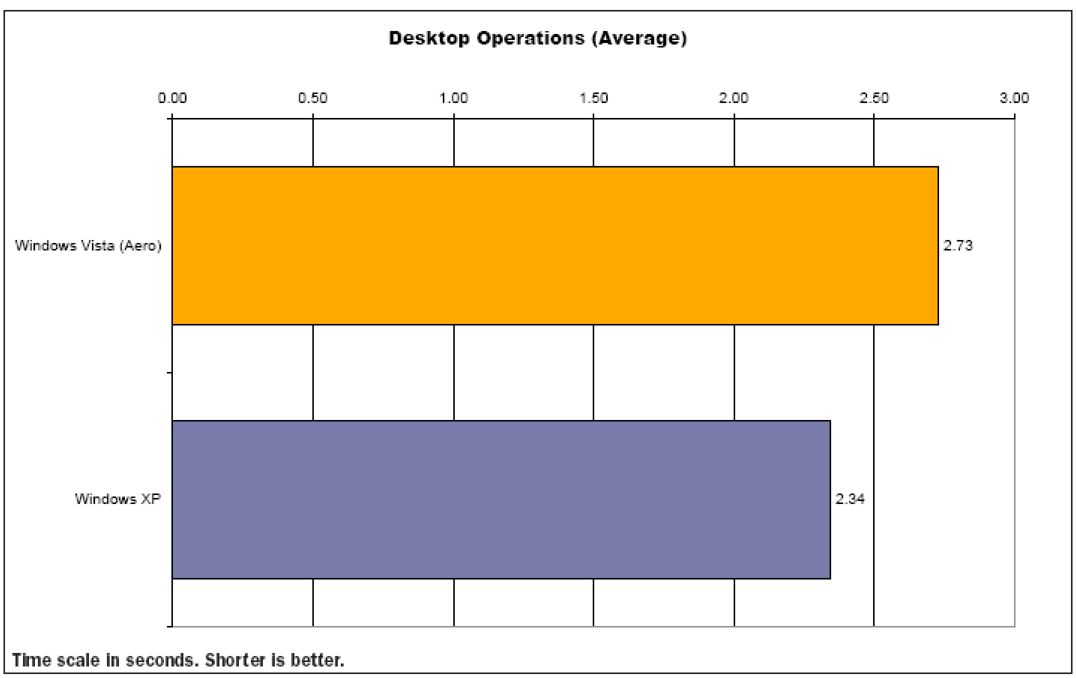

Figure 8. Desktop Operation Average (From Pfeiffer Consulting, 2007)

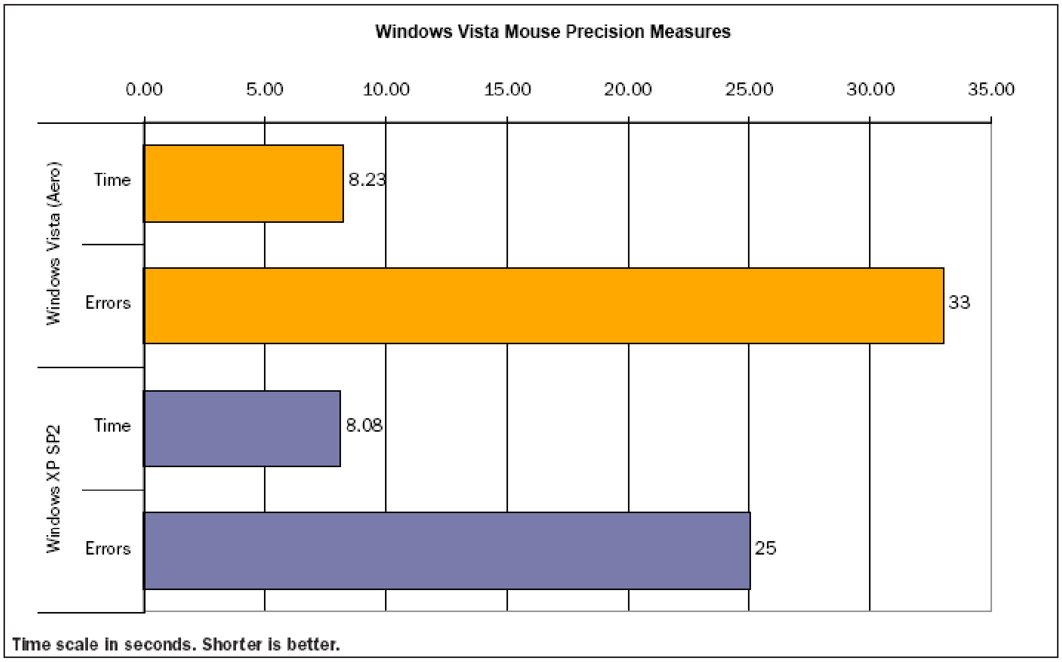

Figure 9. Mouse Precision Measures (From Pfeiffer Consulting, 2007)

H. WINDOWS 7

The architecture of Windows 7 is built on same foundation as Windows Server 2008 and Windows Vista. This is to ensure that virtually all personal computers running on Windows operating systems, applications and devices that are already compatible with Windows Vista will continue to remain compatible with Windows 7 (Microsoft Corporation, 2008b). It has been touted as the remedy of some of the problems that have plagued Windows Vista, such as cold boot attack and memory protection weaknesses. The minimum requirements for Windows 7 are:

- 1GHz processor (32- or 64-bit)
- 1GB of main memory
- 16GB of available disk space
- Support for Direct X 9 graphics with 128MB of memory
- A DVD-R/W drive (Microsoft Corporation, 2008b).

Significant improvements of Windows 7 include less clutter on the desktop, ranging from the elimination of the gadget sidebar on the right side of the screen, to the

eradication of the constant stream of balloon alerts that pop up from the task bar (Sangani, 2009a). And it is because of these problems with Vista that many small businesses have decided to keep Windows XP. Microsoft's solution was to create a backwards compatibility mode for Windows 7 for XP users. XP Mode will be available to users of the Windows 7 Professional, Ultimate, and Enterprise versions – the first two being the highest-priced versions of Windows 7, with Enterprise being sold only through volume licensing agreements (Sangani, 2009b).

One improvement of note is the increased speed that Windows 7 has over its predecessors. One can find various benchmarks performed on Windows 7 all over the internet with most praising the overall improvements over Windows Vista. ZDNet performed a series of benchmarks comparing Windows 7, Vista SP2, and Windows XP SP3. For their benchmark, they used the test platforms described in Figure 10.

Test platforms			
	Low end	Mobile	High end
System	MSI Wind NT 3325	Acer Timeline 3810T	Custom made
Processor	Atom N330	Core Solo U3500	Core i7 965
CPU type	dual core	single core	quad core
Clock speed	1.6GHz	1.4GHz	3.2GHz
Graphics card	Intel GMA 950	Intel GMA 4500	ATI Radeon HD 4870
Drivers	15.12.75.64.1825	Windows 7	ATI Catalyst 9.6
WDDM version	1.0	1.1	1.1
Motherboard	OEM	OEM	Gigabyte EX 58DS4
Memory	2GB (DDR2)	4GB (DDR2/533)	6GB (DDR3/1600)
Storage	500GB WD HD	256GB Samsung SSD	256GB Samsung SSD

Figure 10. Windows 7: Test Platforms (From Smith, 2009)

The startup and shut down times were recorded for the three operating systems on the high-end test. Two separate startups were conducted; one to measure the time to reach the desktop and the second to launch Internet Explorer and the Microsoft Bing homepage. Windows 7 takes the lead with a faster start up to open the desktop and launch IE and Bing at 12 and 14.5 seconds, respectively. Vista and XP both took a little over 14 seconds to reach the desktop from startup and launch IE and Bing at 18.5 and 23.7 seconds, respectively. To shut down, Window 7 demonstrates a faster time by

shutting down in 4.5 seconds compared to Vista 7 seconds and XP 6.5 seconds (Smith, 2009). Figure11 illustrates the results of these benchmarks.

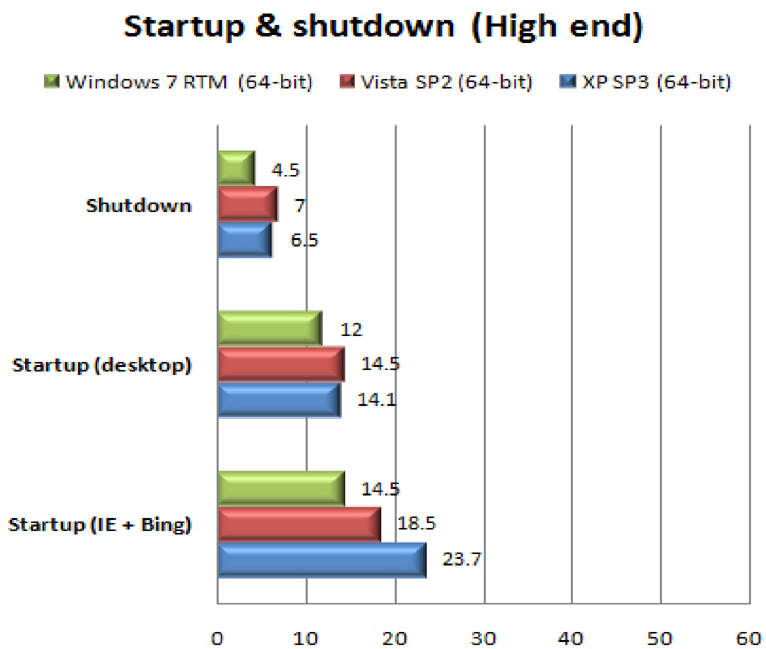

Figure 11. Startup and Shutdown–High End (From Smith, 2009)

On the low-end test, the benchmark times were longer. Windows 7 outperformed XP during the startup tests. During shutdown, XP was faster by one second. Figure 12 illustrates the results.

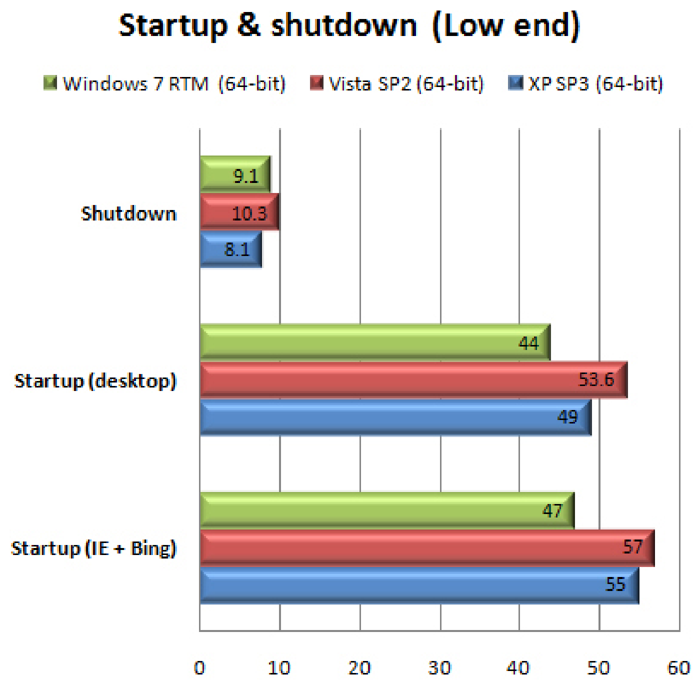

Figure 12. Startup and Shutdown–Low End (From Smith, 2009)

THIS PAGE INTENTIONALLY LEFT BLANK

III. LICENSING CONSIDERATIONS

Before a company installs and uses software or tools, such as Windows Server 2008 or 2003, an agreement between the company and the software manufacturer must be made. The software license is the single most important contract that the company will sign in a packaged software implementation project and is the one contract that will be included in every project (Tayntor, 2006). This essentially allows the company the legal rights to be able to use a vendor's proprietary product.

Licensed software should include any Updates and Upgrades along with the software that was purchased with the original product. This is to ensure that the company has a right to use all versions of the software that the vendor provides (Tayntor, 2006). The licensee should include any subdivisions or affiliates in their agreement to use the licensed software. This will prevent any additional licensing fees or agreements to be negotiated.

A. LICENSING (WINDOWS SERVER 2003)

In a Terminal Services environment, one license for each server is required in order to run the Microsoft product. This license, known as a "server license," is just the standard Windows Server 2003 license; you do not need anything special to run Terminal Server (Madden, 2004). In order to implement a terminal server, a license for each server and client is required. Microsoft has three options for licensing Terminal Server clients:

1. Terminal Server "Device" Client Access License

Terminal Services licensing has traditionally been handled by the Terminal Server device Client Access License. One license is assigned to each specific client device. Each unique client device that accesses a Terminal Server requires a single TS Device CAL (Madden, 2004). This license is best used for multiple users in 24-hour work environments such as hospitals and call centers. Users can share a single TS Device CAL.

2. Terminal Server "User" Client Access License

A Terminal Server user Client Access License is assigned to a user account. It then "follows" that user no matter which server he logs on to and no matter which client device he logs on from (Madden, 2004). This option best suits users that go from one location to another using TS to access their applications or have multiple client devices.

3. External Connector License

This option is designed to be used when systems are extended to external parties, including business partners and the public (Madden, 2004). The ECL is a way to connect your server and provide a concurrent user license to outside users.

4. Microsoft Windows Server Client Access Licenses

To legally access a Windows 2003 Terminal Server, each client seat requires each of the following licenses:

- Windows Server 2003 Client Access License. This is required for any user that needs access to Windows Server 2003.
- Windows Server 2003 Terminal Server Client Access License. This builds upon the regular Windows Server CAL, adding the legal right for users to access a "remote control" session on a Terminal Server; if you have a 5000-user Active Directory environment with a few Terminal Servers that provide applications for 300 users, you'll then need 5000 Windows Server CALs and 300 Terminal Server CALs (Madden, 2004).

5. Licensing Components

There are four components that make up the Terminal Services for Windows Server 2003 as illustrated in Figure 13.

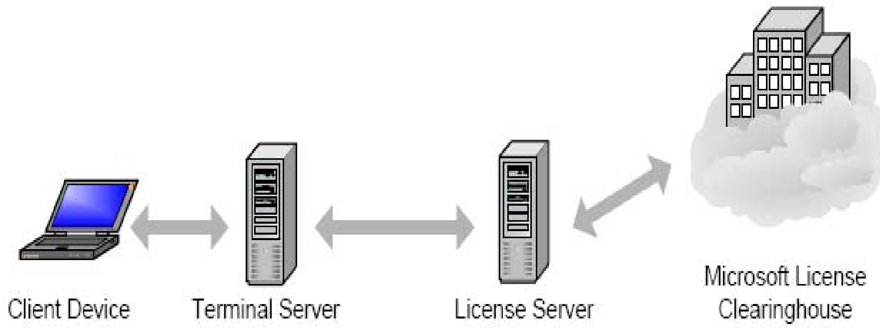

Figure 13. Licensing Components (From Madden, 2004)

- Terminal Services licensing servers: This comes standard and already installed with Windows Server 2003. Digital certificates for TS CALs are stored in this license server that is distributed to client devices.
- The Microsoft license clearinghouse: Prior to use, products licensed by Microsoft must be activated through the Microsoft license clearinghouse. The Microsoft license clearinghouse is a large Internet-based certificate authority that authorizes and activates these licenses and servers (Madden, 2004). This is to ensure that their products are not stolen or illegally copied and pirated. Any TS license that is not activated by the clearinghouse will only be effective for 90 days before the TS CALs expire.
- Windows 2003 Terminal Servers: Communication between the terminal server and the client is conducted to ensure that client devices are licensed.
- Licenses: There are several different licenses that run on Windows Server 2003 depending if it is for a per user, per device, or external connector license.

6. Deploying Windows Server 2003 Terminal Services

Windows Server 2003 licenses will only work with a Windows server 2003-based terminal server license server. It will not function with Windows Server 2000 unless there is an upgrade of an existing Windows Server 2000 License Server. Windows Server 2003-based terminal servers can automatically discover a Terminal Server License Server installed on a member server running Windows Server 2003 configured as an Enterprise License Server in the Active Directory services site (Madden, 2004).

B. LICENSING (WINDOWS SERVER 2008)

1. Windows Server 2008 Requirement

Consistent with Windows Server 2003 R2, the licensing of Windows Server 2008 requires the purchase of a Windows Server 2008 Client Access License. A Terminal Services 2008 CAL is required, in addition to a Windows Server 2008 CAL, to use the Terminal Services functionality of the server software; similarly, a Rights Management Services 2008 CAL is required, in addition to a Windows Server 2008 CAL, to use the Rights Management Services functionality of the server software (Microsoft Corporation, 2007a). External Connector license is also available for external user access instead of individual CALs. The decision trees illustrated in Figures 14 and 15 help determine if a CAL or External Connector is required.

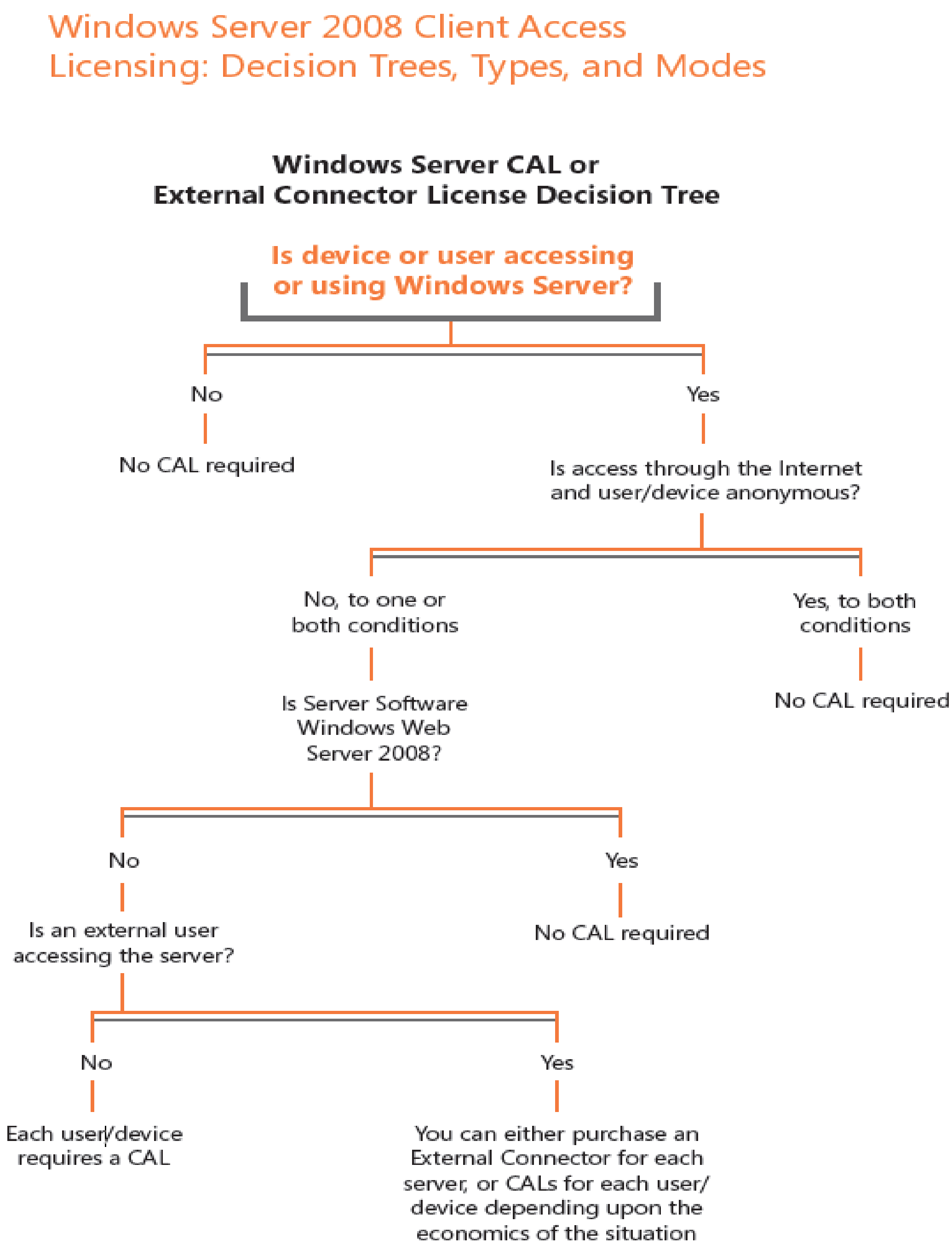

Figure 14. CAL Decision Tree Windows Server 2008 (From Microsoft Corporation, 2008a)

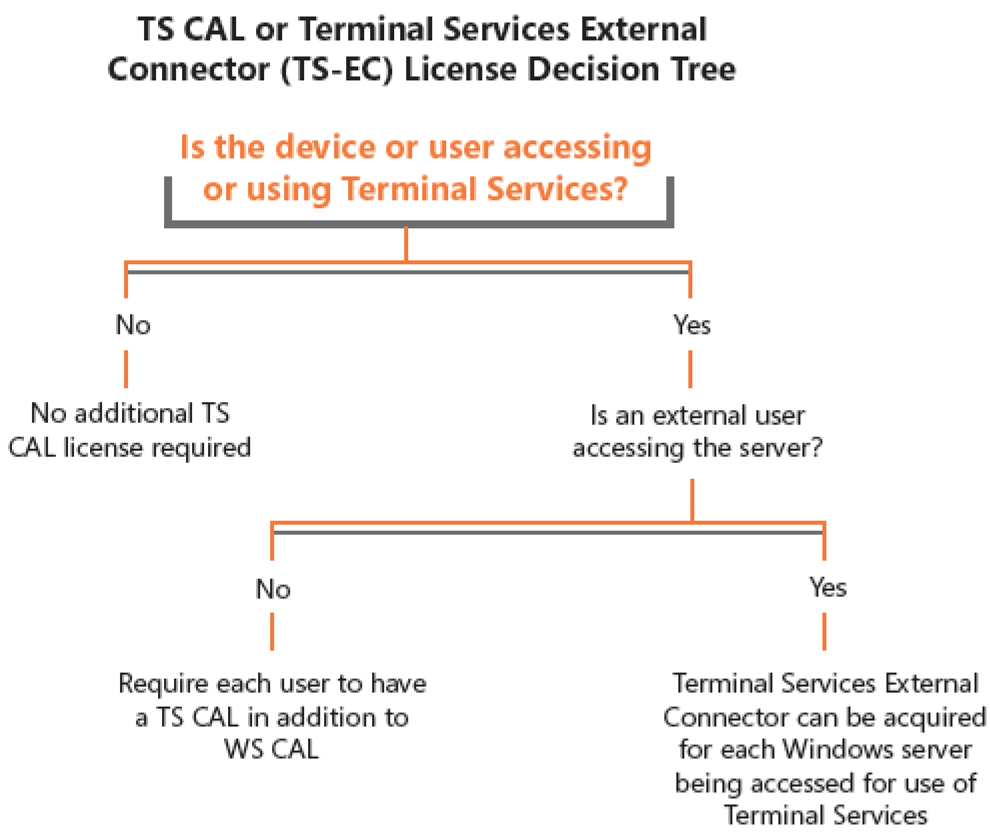

Figure 15. Terminal Services License Decision Tree Server 2008 (From Microsoft Corporation, 2008a)

Figures 16 and 17 describe the different CAL types and modes:

CAL Types

Device CAL	User CAL
Acquire an access license for every device (used by any user) accessing your servers.	Acquire an access license for every named user accessing your servers (from any device).
Makes the most economical and administrative sense for an organization with multiple users for one device, such as shift workers.	Makes the most economical and administrative sense for an organization with many roaming employees who need access to the corporate network from unknown devices, and/or employees who access the network via multiple devices (that is, if the number of users is less than the number of devices, then this is the more economical choice).

Figure 16. CAL Types (From Microsoft Corporation, 2008a)

CAL Licensing Modes

Per User or Per Device Mode	Per Server Mode
The number of Windows CALs required equals the number of users or devices accessing the server software (the number of servers accessed does not matter).	The number of Windows CALs required equals the maximum number of users or devices that may simultaneously access or use the server software running on a particular server. The Windows CALs you acquire are designated for use exclusively with a particular server.

Figure 17. CAL Licensing Modes (From Microsoft Corporation, 2008a)

In order to access Windows Server 2008 software, Windows Server 2008 CALs must be purchased. Windows Server 2003 CALs will only run with Windows Server 2003 unless a Software Assurance upgrade had been purchased. If Software Assurance is purchased for Windows Server 2008, the CALs can then be used for the next Windows Server upgrade. If your Windows CALs are covered under Software Assurance, you can convert those Windows CALs from Device CALs to User CALs, or vice versa, when you renew your Software Assurance. If your Windows CALs are not covered under Software Assurance, you may not switch; this also applies to TS CALs (Microsoft Corporation, 2007a).

2. Terminal Services Licensing Requirements

Terminal Services functionality in Windows Server 2008 lets you remotely execute applications on a Windows-based server from a wide range of devices over virtually any type of network connection. A server running Terminal Services can be referred to as a Terminal Server (Microsoft Corporation, 2007a).

3. Windows Server 2008 External Connector Licenses

If you would like to allow your business partners or customers to access your network, and do not want to purchase individual CALs for each of them, you can acquire a Windows Server 2008 External Connector license for each Windows server that will be accessed by these external users (Microsoft Corporation, 2007a). Windows servers that

will be accessed by an external user will require an EC license and when a Terminal Services EC license is acquired. A Windows Server 2008 EC license can be used on a Windows Server 2003 licensed server.

4. Windows Server 2008 with Hyper-V Technology

The four main products of Windows Server 2008 (Standard, Enterprise, Datacenter, and Web Server) all come with Hypervisor (Hyper-V) Technology. A key feature in the Windows Server 2008 core operating system, customers can choose not to have this technology. Customers who choose to purchase Windows Server 2008 products without Hyper-V will need to separately license the hypervisor technology, whether it is Microsoft Hyper-V, Microsoft Virtual Server R2, or a third-party hypervisor technology (Microsoft Corporation, 2007b). Otherwise, the licensing terms are exactly the same.

IV. BENCHMARK TEST ENVIRONMENT AND DESCRIPTIONS

A. INTRODUCTION

The performance test tools utilized in this project were either freeware utilities or native Microsoft Windows tools. This level of availability allows workflow and performance tests to be easily repeatable. The tools utilized were OfficeBench 7, PerformanceTest 7.0, NovaBench 3.0.1, and Microsoft Corporations Reliability and Performance Monitor Version 6.0.6002. These performance test utilities cover a large area of system tests, which are primarily focused on time-to-completion of tasks. Each of the tools was critical in producing a well-researched decision as to which operating system(s) is optimal for the GSBPP.

B. CLIENT-SERVER ARCHITECTURE AND DESIGN

The client–server architecture is a structure that divides tasks between a provider of resources (the server) and resource or service requesters (clients). A server acts as the host of several applications or data. In turn servers allow clients to share their resources. The client’s function is to request a server's data or application functions. Clients generally initiate communication with servers who sit by patiently listening for these requests. The idea of this architecture is to provide multiple users with access to the same resources or data (Reese, 2000, p. 128–129).

1. Server Specifications

To ensure the performance tests in this project were based on an equal baseline we used one custom built machine to host four server operating systems. The specifications of the computer used as the server are as follows:

- Motherboard: ASUSTeK P6T Deluxe V2
- Chipset: Intel X58
- CPU: 1 X Intel Core i7 975 @ 3.33 GHz (w/ 4 cores)
- RAM: 12GB (6 X 2046 DDR-3 SDRAM)
- GPU: NVIDIA GeForce 7600 (256MB Memory)

- Network Card: Yukon 88E8056 PCIe Gigabit Ethernet Card
- Hard Disk: WDC (2000GB)

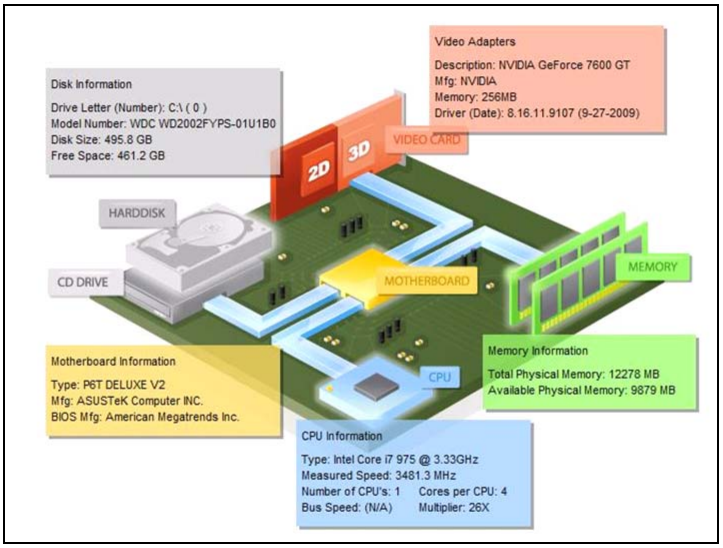

Figure 18. Server Hardware Architecture

a. Server Operating Systems Tested

The hard disk was partitioned into four equal partitions of 500GB. On each partition a separate Windows Server Operating System was installed.

- Windows Server 2008 Enterprise Professional (x64) SP2
- Windows Server 2008 Enterprise Professional (x32) SP2
- Windows Server 2003 Enterprise (x64) SP2
- Windows Server 2003 Enterprise (x32) SP2

Each operating system was immediately patched to the most current set of security and system updates available through Microsoft's update service.

b. Applications and Services

Continuing with our intent to keep an equal performance test structure each partition was loaded with a minimal amount of applications and services. All four servers were loaded with the following:

- Internet Explorer Version 8.0.7600
- Microsoft Office 2003 Professional
- Microsoft Office 2007 Enterprise
- OfficeBench 7

- PerformanceTest 7.0
- NovaBench 3.0.1
- Stata I/C 11 (64- / 32-bit versions, respectively)
- Terminal Services
- Domain Name System (DNS)

2. Client Architecture

The same workflow was used in designing the client architecture. One laptop computer was used to host both of the client operating systems. The specifications of the computer used as the client are as follows:

- Computer Make and model: Dell Latitude D620
- Motherboard: Dell 0TD761
- Chipset: Intel 479 mPGA
- CPU: 1 X Intel CPU T2600 @2.16GHz
- RAM: 4GB (DDR-2 667)
- GPU: Intel Calistoga Graphics Controller (224MB Memory)
- Network Adapter: Dell Wireless 1490 Dual Band WLAN Mini Card
- Hard Disk: Hitachi (80GB)

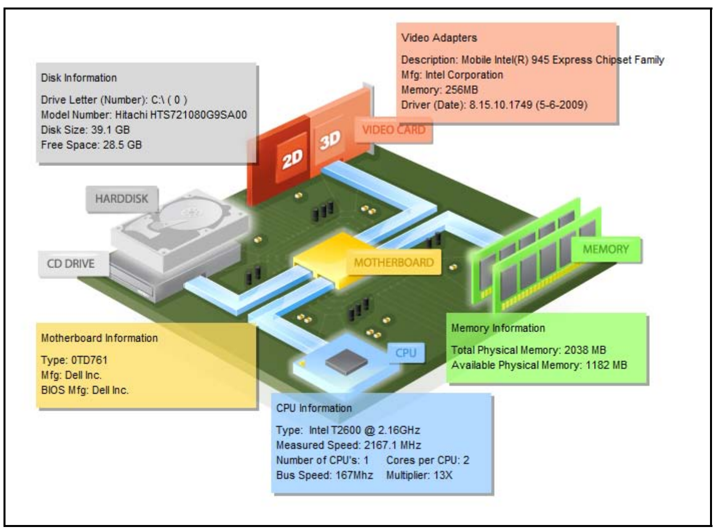

Figure 19. Client Equipment Architecture

a. *Operating Systems*

The hard disk was partitioned in to two equal partitions of 30GB. On each partition a separate Windows Operating System was installed.

- Windows XP Professional SP3
- Windows 7 Professional

Once again each operating system was immediately patched to the most current set of security and system updates available through Microsoft's update service.

C. CLIENT-SERVER COMMUNICATION DESIGN

In order to perform several of the benchmarks a client-server relationship was established using Terminal Services and Domain Name System services.

Terminal Services is a role that allows remote computers to run desktops and applications on a server as though it performing those actions locally. Client input (keystrokes and mouse instructions) is sent to the server over a network and a visual display of the information is sent back to the client (Allen, 2001). Advantages of using Terminal Services are:

- Because the work is performed on the server, the client does need to be powerful.
- Administration of applications is centralized and easier to manage.
- Users on the client computers do not have the ability to misconfigure their computer, since configuration is handled in large on the server (Allen, 2001).

The Domain Name System is a distributed Internet directory service that helped our ad hoc network to translate IP addresses into human friendly domain names (Salamon, 2008). Since DNS was used, static IP addresses were assigned to this project via the Information Technology Assistance Center (ITAC) at NPS.

User accounts and passwords were established with full administrator rights and communication was finalized by using Remote Desktop Connection from the client side.

It is worth noting at this point that set up of the ad hoc networks went especially well when using Windows Server 2008 (both 64- and 32-bit versions). Setup was done via a wizard that led us through the steps needed. However, set up of the ad hoc network

proved to be much more challenging when using Windows Server 2003 (both 64- and 32-bit versions). In order to establish the client-server communication we were forced to make a concession and disable the native firewall applications.

D. PERFORMANCE TEST TOOLS USED

At this point, we will now turn our attention to which benchmark tools we used, how they were set-up and the workflow adopted for each set of performance tests. The benchmark tools we used were:

- OfficeBench 7
- PerformanceTest 7.0
- NovaBench 3.0.1
- Microsoft Reliability and Performance Monitor
- Stop watch

E. PERFORMANCE TEST I: OFFICEBENCH 7

Our goal was to assess the relative performance of varying flavors of Windows Server ((64-bit) and (32-bit) versions of Windows 2003 and Windows 2008 Server). Each operating system was stressed to record the time required to access, retrieve, and create documents in Microsoft Office 2007 and 2003. Methodical application of the benchmarking tool was essential to gathering quality results. Once again, time-to-completion speed was the priority and the metric gathered was how quickly the benchmark completed its test script. This test does not account for the levels functionality of the systems, simply the speed at which the benchmark was complete.

1. Overview

OfficeBench 7 benchmark was created by Mr. Randall C. Kennedy of Competitive Systems Analysis, Inc. The script was originally designed to benchmark the Pentium III and Pentium 4 CPUs (Kennedy, 2009a).

OfficeBench is described as a cross-version test script, which uses Microsoft Office to evaluate computer performance. It is designed to run anywhere, meaning that the script will execute reliably under almost any Windows operating system (Kennedy,

2009a). It can also run multiple versions of Microsoft Office; which made it an exceptional tool for testing commonly used business applications at GSBPP. OfficeBench measures the performance metric of time-to-completion of standard business processes done in Word, Excel, PowerPoint, and Internet Explorer. OfficeBench 7 drives its script with Object Link Embedded (OLE) Automation fostering the benchmarking to run unmodified across varying Microsoft Office versions and different Operating Systems (Truttman, 2009). The tool's ability to run unmodified was essential since variance in the behavior of the benchmarking tool across Microsoft Office versions and operating systems would skew the results obtained from testing (Kennedy, 2009a).

2. Workflow and Challenges

In order to perform an accurate analysis of the four Operating Systems, it was important to maintain a precise and consistent workflow process during the testing procedures. This would put all of the benchmarks on equal footing and allow an unbiased view of which system and which software suite was the best combination.

For each test of the system, it was important that system memory be as unencumbered as possible. We ensured this by performing a restart of the system whenever a test was to be performed. Even though this added to the workflow time, it was an important step. Running the benchmark program without restarting the system presented a three-fold increase in benchmark times in some instances.

In addition to restarting the system, we also ran a single loop test on each operating system in order to get a measurable baseline. For the actual test, we used a ten-loop sequence. These options were built into the benchmarking program. Any number of loops could be performed. For time sake, we stopped at ten loops.

An interesting challenge arose during this particular set of benchmark tests. The Office software suites did not function together very well. Leaving both Office suites installed on the same partition caused OfficeBench to randomly use Office 2003 or Office 2007 to perform the tested tasks. In order to ensure that results were accurate and that we were getting results for correct applications, we had to uninstall the Office suite we were not testing. To be exact when the OfficeBench tests were finished on Microsoft

Office 2003 we uninstalled them and installed Microsoft Office 2007. Only then did OfficeBench perform the script flawlessly.

3. Benchmark Description

OfficeBench is a fairly complex benchmark tool. It in essence takes typical business software (Microsoft Office and Internet Explorer) and runs a series of actions and measures performance. OfficeBench leverages Object Link Embedded automation to run applications. Using OLE automation is what allows us to run unmodified across Office versions. It also mitigates anomalies that may inhibit the script (Kennedy, 2009b). This script does the following tasks automatically:

- Reformat all section headers and subheads in a Word document
- Generate multiple chart objects in Excel
- Generate a multi-slide presentation in PowerPoint
- Multi-page scroll with copy and paste of charts into Word
- Slide sort/apply multiple templates in PowerPoint
- Multi-page scroll/print preview/print-to-file in Word
- Multi-chart print preview/print-to-file in Excel
- Global search/replace in Word (multiple times)
- Multi-slide preview/print-to-file in PowerPoint
- Navigate simulated research web site in IE (multiple) (Kennedy, 2009b).

The individual tests are then consolidated in to a graph; generally, the smaller the bars the better the performance.

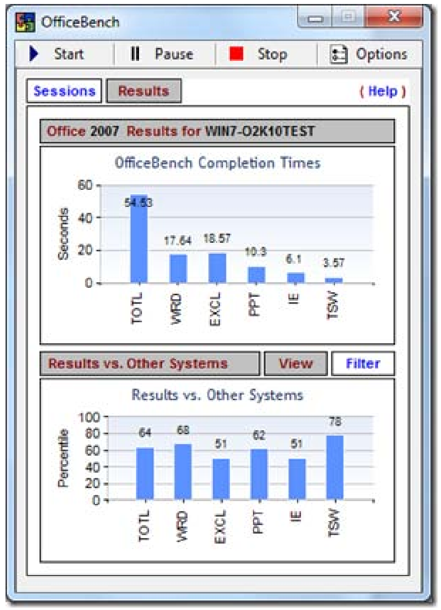

Figure 20. OfficeBench Screen Capture

The example in Figure 20 is not an actual result; results for this all other benchmarks will be listed in results portion of this document.

F. PERFORMANCE TEST II: PERFORMANCETEST 7.0

1. Overview

PerformanceTest 7.0 was created by PassMark Software Pty Ltd, which is a software development company founded in 1998; located in Sydney, Australia (Wren, 2005).

This benchmarking tool is made to rapidly assess the performance of a personal computer and compare it to other baseline computer systems (Robinson, 2010, p.1). However, for our purposes we simply used the results to compare them against each of our Windows Server operating systems. Should there be interest in reviewing the baselines, they are available through PerformanceTest or from PassMark's Web site (http://www.passmark.com) (Wren, 2008).

Originally, the program was designed to be run from a CD or USB memory stick to allow users a tool to use while purchasing a PC. This means you can take it with you when you go out shopping for a new or second hand PC (Robinson, 2008). However, for our purposes, we simply installed the application across all four operating system environments.

2. Workflow and Challenges

A similar workflow to the one used in the OfficeBench tests was utilized. After the installation of PerformanceTest on each partition of the server machine, we rebooted the system and allowed ample time for all startup applications and services to execute. We then ran the scripted test and recorded the results.

When testing each version of Windows Server 2003 we found that there was an issue with the 3D graphics script. We ensured that the most current driver available for our graphics card was loaded and that the correct version(s) of DirectX were installed. We noticed that when using the DirectX Diagnostic snap in (dxdiag) that 3D acceleration was unavailable. We were not able to resolve this issue so the overall combined scores will be skewed. However, it is important to note that the individual scores that are not dependant on DirectX such as all of the CPU tests and the 2D benchmarks; those two areas are still valid and useful in this study.

3. Benchmark Description

The PerformanceTest standard test suite consists of twenty-eight standard benchmark tests. The standard suite tests are consolidated in to the following groups.

- CPU tests Mathematical operations, compression, encryption
- 2D graphics tests Drawing lines, bitmaps, fonts, text, and GUI elements
- Memory tests Allocating and accessing memory speed and efficiency
- 3D graphics tests, DirectX 3D graphics and animations
- Disk tests Reading, writing and seeking within disk files
- CD / DVD test on the speed of a CD or DVD drive (Robinson, 2008).

These tests are run via a script to ensure an equal baseline and the timing for these tests is done via high resolution timers and are accurate to 1 millionth of a second (Robinson, 2010, p. 2).

The results are then combined into an amalgam rating entitled a "PassMark rating". The following screenshot in Figure 21 of a PassMark rating is only an example, not a systems actual score.

Figure 21. PassMark Rating Screen Capture

Despite the problems with DirectX affecting the overall PassMark rating, it is still useful to view the initial results since it provides us with a topical indication of how each system performed. A “PassMark rating” is a weighted average of all the other test results, which represents an overall indication of the server machines performance. In this benchmark the bigger the number, the faster the computer is rated. For example, older Intel Pentium 4 machines get a rating of about 800, while a Core2 Duo machine with 4GB of memory typically rate around 1300 (Robinson, 2010, p. 4). The weights of each test are presented in Table 3.

Test Suite	**Weighting**
Disk	21%
CD / DVD	5%
Memory	19%
3D Graphics	12%
2D Graphics	14%
CPU	29%
Total	**100%**

Table 3. PassMark Rating Weighted Averages (From Robinson, 2008)

> These weighting's are based on the "average" computer usage and are intended to give the user of PerformanceTest an overall indication about how this computer will perform in general use. The score is also calculated in such a way that a single extremely high value cannot significantly improve the final score. All components in a system must be performing well in order for the final score to be high. To do this, each sub section score has the following formula applied to it.
>
> Weighting equation: (e.g. 0.29 for the CPU) / sub-score.
>
> Once each sub-score has this formula applied to it they are then added together. This produces a number that gets smaller the faster the system is. Finally, in order to get a number that gets bigger as the system improves the number is inverted again (i.e. 1 / sub-score-total). (Robinson, 2010, p. 4)

Additionally, results can be drilled down into and are presented in bar charts. These bar charts are what we will base our analysis on.

Results for this and all other benchmarks to be listed in “Results” portion of this document.

G. PERFORMANCE TEST III: NOVABENCH 3.0.1

1. Overview

NovaBench was developed by Nathan LaPierre from NovaTech Network Company in Halifax, Nova Scotia, Canada.

NovaBench is also a free benchmark tool that performs a similar scripted test like the one PerformanceTest executes. It tests CPU speed, performs multi-threaded tests for multiple cores/processors, hardware accelerated graphics tests, and hard drive write speed test (Romero, 2008). With one key difference, it allowed us to turn off all 3D graphics test, which mitigated our DirectX challenges within Windows Server 2003.

2. Workflow and Challenges

Once again, we used a similar workflow to the ones used in the OfficeBench and PerformanceTest. We installed NovaBench on each partition of the server machine, rebooted the system and once allowed ample time for all startup applications and services to execute. We then ran the NovaBench script with the 3D Graphics option deselected and recorded the results.

A disadvantage of NovaBench is the lack of transparency in how the NovaBench score is calculated or weighted. However, in general the higher the score the better / faster the system executed the NovaBench script.

3. Benchmark Description

Our NovaBench performance test consisted of five tests consisting of the following:

- Floating Point Test - Tests CPU's floating point arithmetic speed
- Integer Test - Tests CPU's integer arithmetic speed
- MD5 Hashing Speed - General CPU test
- RAM Speed - Tests RAM read and write speed
- Disk Write Speed - Test write speed of primary device (LaPierre, 2010).

Each test only lasted about 2–3 minutes and produced an overall NovaBench system score and a breakdown of the individual areas listed above. Figure 22 displays a NovaBench result as an example only, not a systems actual score.

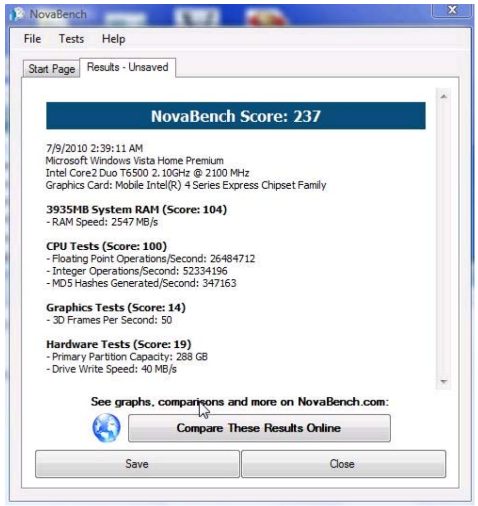

Figure 22. NovaBench Results Screen Capture

Once again, we were presented with the option to compare our server(s) with several baseline systems, but for the purpose of these test we only needed to compare the varying server operating system in our project against one another.

Results for this and all other benchmarks to be listed in “Results” portion of this document.

H. PERFORMANCE TEST IV: SYSTEM START UP AND SHUTDOWN

1. Overview and Challenges

Next, we conducted a time-based test by measuring how long it took for each system to start up and shutdown. These tests were conducted manually using a stopwatch to measure time elapsed.

It is important to note that no instant-updates or applications were running or installing in the background that would impact our results.

However, Windows Server 2008 (64- and 32-bit versions) and Windows Server 2003 (64- and 32-bit versions) run different variations in startup files and services. We

have maintained all along that our goal was to have equal baselines for all operating system environment benchmarks. Despite performing this test with a system that is a close to an “out of the box” installation of the operating system(s) as we could, we do admit that there are some differences in startup files and services that may impact the results of these tests.

Some of examples in the variation of services running are:

- HyperV on the Microsoft Server 2008
- Differing instances svchost.exe processes across all versions

We also conducted a wake up time test where we measured how long it took each system to return from a sleep state. However, all systems took less than 1 second to return from sleeping. As a result, we decided to forgo the test and results since the time was too short to produce meaningful differences in numbers.

2. Benchmark Description

Generally, a benchmark test such as this is done by starting a stopwatch at the same time you turn on your test machine. However, since we have partitioned the hard drive to accommodate for four server operating systems, we began each test from the moment we selected an operating system from boot manager interface. We stopped the clock at the moment we reached the log in screen. We then typed in our credentials and begin the stopwatch again at the same instance we pressed Enter to begin the log in process. We found that discounting any keystrokes was only way to maintain fair startup time results. The stopwatch was stopped again when the first instance of the desktop appeared. The two results above were added together to produce our startup time result.

We then allowed ample time for all services and startup applications to complete their execution before selected the option to shut each system down. We closed any applications that were running. The stopwatch began in tandem with hitting the enter key and stopped when the “ON” light on the CPU turned off.

Results for this and all other benchmarks to be listed in “Results” portion of this document.

I. PERFORMANCE TEST V: SERVER–CLIENT PERFMON TEST USING STATA 11 I/C

1. Overview

Next, we wanted to test how the differing flavors of Windows Server operating systems did when executing another common every day computing task(s) performed at the GSBPP. We chose a statistics application that is quite commonly used named Stata 11 I/C. StataCorp produces Stata, which is very well respected company in statistical software arena. Stata provides an integrated statistics, graphics, and data-management solution for anyone who analyzes data (Webmaster, 2010).

We performed a number of choreographed Stata tasks and measured the efficiency of selected system resources using the native Microsoft Corp. Reliability and Performance Monitor snap in (PERFMON).

It was also one of our secondary goals of the project to test two variations of client operating systems to gain some insight as to how different clients might affect server performance. For the purpose of this test, we used:

- Windows XP Professional SP3
- Windows 7 Professional

2. Workflow and Challenges

Stata 11 I/C was installed and configured on each partition of the server machine and then each was rebooted to allow for as clean a state as we could achieve across each environment. Each client was then booted up and was allowed ample to finish the execution of any start up programs and services.

Next we established communication between the server and the client machines using REMOTE DESKTOP CONTROL. We were then prompted for our login credentials and from this point forward in the test all keyboard and mouse input was done from the client, which was requesting resources and services from the server.

From the client we then called up the Performance Monitor snap in from the RUN command line using the "PERFMON" command. Once Performance Monitor was running we started up Stata and opened a relatively large database file (27.5MB) consisting of military personnel demographic data. We then organized the data by the ascending order of "RACE". Next we produced a "PIE CHART" from the data using the "RANK" variable. Upon completion of the PIE CHART graphic we stopped the performance monitor, took a screen shot using the PRTSCRN key, opened the .BMP in Microsoft Paint, cropped the image and finally named the result according to the operating systems tested.

It is important to note that we did not use a peer-to-peer client to server connection; instead we chose to use the NPS Network as our connection. This was done largely in part to provide the team with the experience of learning how to set up an ad hoc network. However, we do recognize that any usage of the NPS Network will slightly skew our results. To mitigate this effect we performed all eight instances of this benchmark on a Sunday afternoon when usage of the campus' network would be minimal.

Also, due to our inexperience in building scripts to automate the tasks performed in Stata, there may be some impact to the results from human interaction (key presses and mouse movement). To mitigate this impact our actions were well choreographed and planned out to be as efficient in our executions of the tasks as possible.

Lastly, due to the size of the database file used we had trouble opening the file initially in Stata. To fix this we modified the Stata memory cap from the default 10M to 40M. The Stata website warns: "Keep in mind that requesting more memory than you need can slow your job down and affect other users."

We used the "SET MEM #K|M, G" command to change the amount of memory available to Stata; this was performed at the Stata Command line.

Example command: set mem 40M, permanently.

3. Benchmark Description

We selected three Performance Monitor counters for this test. By default, these counters are selected in the Windows Server 2003 variations, but in Windows Server 2008 they must be manually selected. Regardless, the measurements of these counters are applicable to this test and as a result we used them. The counters used and their descriptions are as follows:

> % Processor Time is the percentage of elapsed time that the processor spends to execute a non-Idle thread. It is calculated by measuring the percentage of time that the processor spends executing the idle thread and then subtracting that value from 100%. (Each processor has an idle thread that consumes cycles when no other threads are ready to run). This counter is the primary indicator of processor activity, and displays the average percentage of busy time observed during the sample interval. (Microsoft Corporation, 2008d, p. 4–5)

> Pages/sec is the rate at which pages are read from or written to disk to resolve hard page faults. This counter is a primary indicator of the kinds of faults that cause system-wide delays. It is the sum of Memory\\Pages Input/sec and Memory\\Pages Output/sec. It is counted in numbers of pages, so it can be compared to other counts of pages, such as Memory\\Page Faults/sec, without conversion. It includes pages retrieved to satisfy faults in the file system cache (usually requested by applications) non-cached mapped memory file. (Microsoft Corporation, 2008d, p. 5)

> Avg. Disk Queue Length is the average number of both read and write requests that were queued for the selected disk during the sample interval. (Microsoft Corporation. 2008d, p. 5)

In general, the shorter and the lower the frequency of spikes in the PERFMON results graph the more efficient the use of the resources measured is. Figure 23 is an example PERFMON results graph.

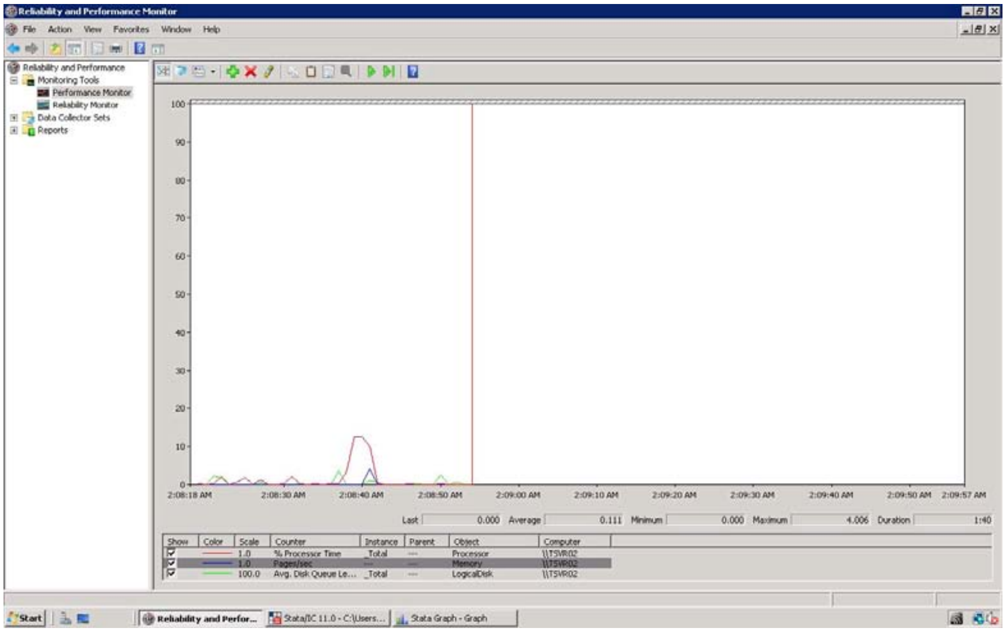

Figure 23. PERFMON Results Graph Screen Capture

Results for this and all other benchmarks to be listed in results portion of this document.

J. PERFORMANCE TEST VI: LIST COMMAND–TIME TO COMPLETION

1. Overview

Finally, we conducted a test using STATA that measured a very relevant task to the GSBPP. We conducted a manual time-to-completion benchmark running a basic LIST command, leveraging a 27Mb file. The LIST command took about 20–40 minutes, respectively, which we found was an adequate amount of time to measure and compare.

2. Workflow and Benchmark Description

We began our test with a freshly booted client and server. The workflow was as follows:

- Using STATA; open the test data file
- Type LIST in the command line (start the clock at the same time)

- The stopwatch was stopped when the red STOP sign at the top of the menu bar turned grey (this indicates the command has executed)
- Stop the clock in tandem with step 4

THIS PAGE INTENTIONALLY LEFT BLANK

V. PERFORMANCE TEST RESULTS

A. INTRODUCTION

The results of the benchmark tests have been grouped in to one section in this document so that they can be easily compared and analyzed by readers. Where graphs or tables enhance analysis, we have used them in this document.

B. PERFORMANCE TEST RESULTS I: OFFICEBENCH

A total of eight performance tests were conducted on the server machine with different combinations of Microsoft Operating Systems and Microsoft Office Programs. The combinations tested were:

- Windows Server 2008 (64-bit) with Office 2007 Enterprise
- Windows Server 2008 (64-bit) with Office 2003 Professional
- Windows Server 2008 (32-bit) with Office 2007 Enterprise
- Windows Server 2008 (32-bit) with Office 2003 Professional
- Windows Server 2003 (64-bit) with Office 2007 Enterprise
- Windows Server 2003 (64-bit) with Office 2003 Professional
- Windows Server 2003 (32-bit) with Office 2007 Enterprise
- Windows Server 2003 (32-bit) with Office 2003 Professional

Figure 25 illustrates the combined results from the benchmark testing. All times recorded for the respective combinations were recorded after the 10-loop run was completed, some with drastically varying results. Windows Server 2008 (64-bit) and Windows Server 2008 (32-bit) had the two closest times for both Office programs, with the loops being completed in approximately 26 seconds and 28 seconds for Office 2007 Enterprise, and roughly 17 seconds and 18 seconds for Office 2003 Professional, respectively.

The most drastic and time-consuming 10-loop combinations belonged to Windows Server 2003 (32-bit), with Office 2007 Enterprise being completed in 35 seconds and Office 2003 Pro concluding the test in 23 seconds. Unquestionably, the fastest test combinations in our benchmark were those utilizing Windows Server 2003

(64-bit). Windows 2003 (64-bit) with Office 2007 Enterprise had results in the 21 seconds range. Windows 2003 (64-bit) also had the best time when using Office 2003 Professional, completing its 10-loop run in an astonishing 11 seconds.

The shorter the bar in the graph the faster a system did in executing the OfficeBench scripted events.

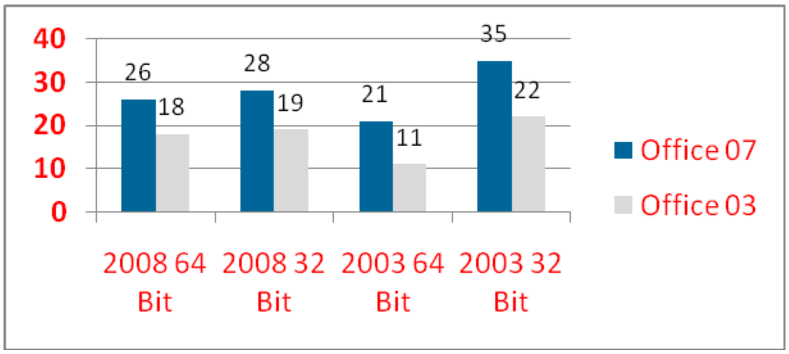

Figure 24. Overall Benchmark Results–Measured in Seconds

1. OfficeBench Drilldowns

Figures 26–29 offer further insight as to how each individual operating system performed in individual areas. The individual areas measured are:

- Total - A total score
- WRD - Result for Microsoft Word
- EXCL - Result for Microsoft Excel
- PPT - Result for Microsoft Power Point
- IE - Internet Explorer
- TSW - Combined results for print to file operations

2. Individual OfficeBench Result Graphs

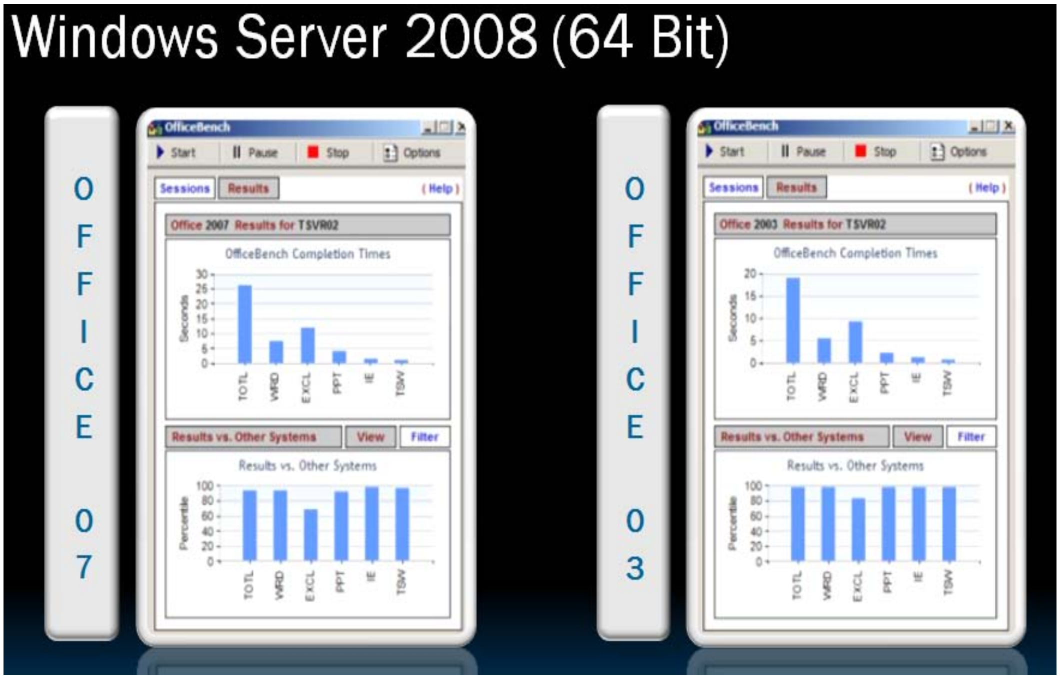

Figure 25. 2008 (64-bit) Results

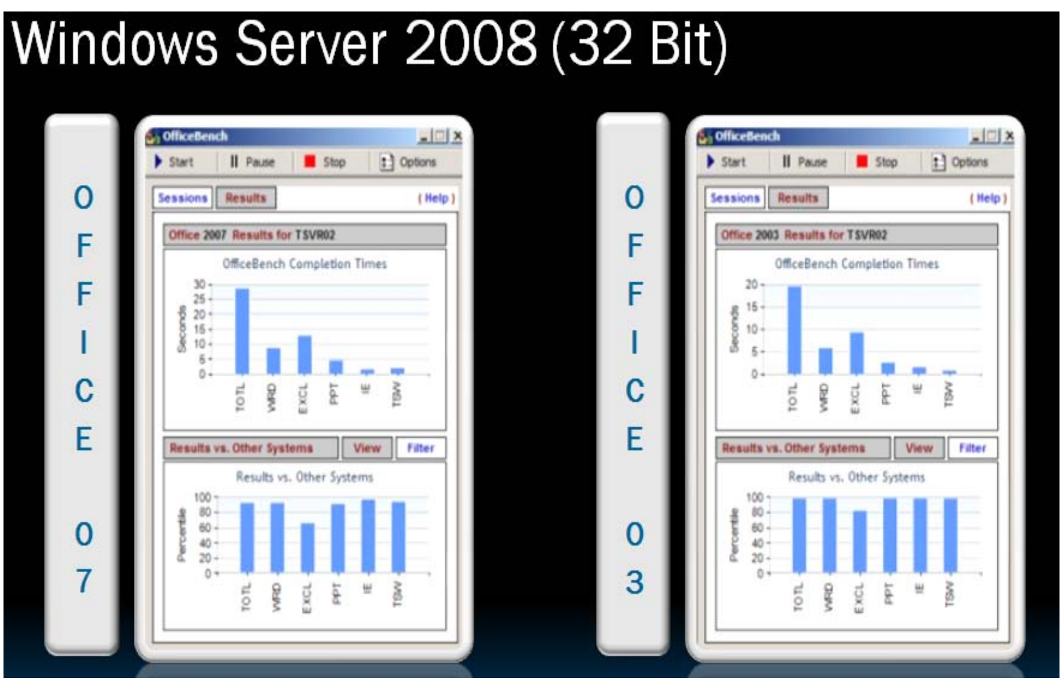

Figure 26. 2008 (32-bit) Results

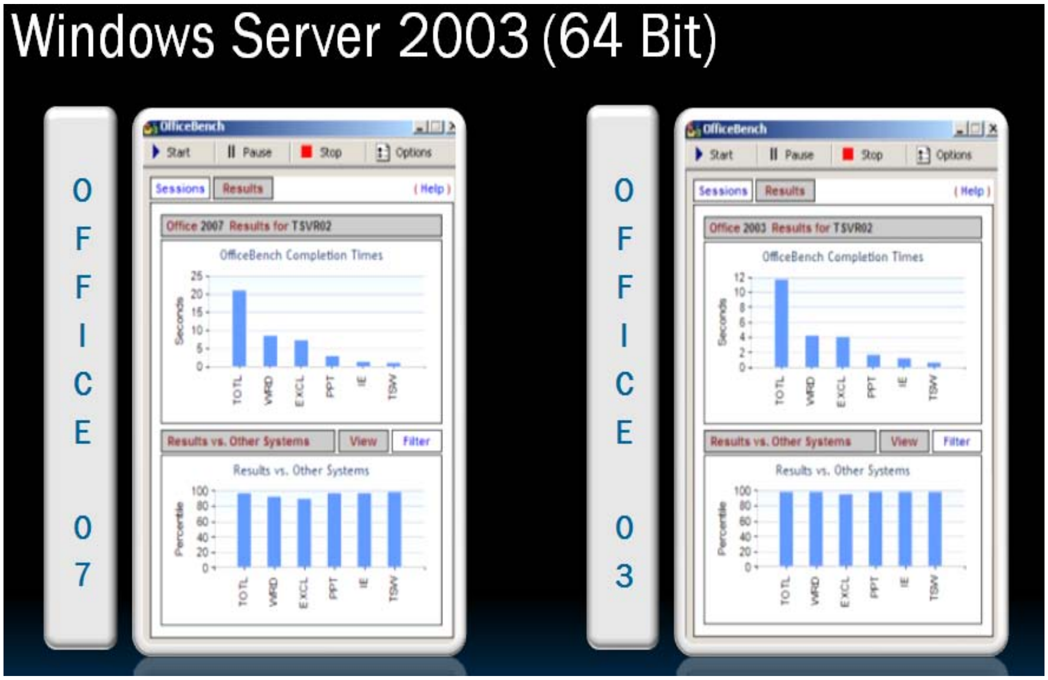

Figure 27. 2003 (64-bit) Results

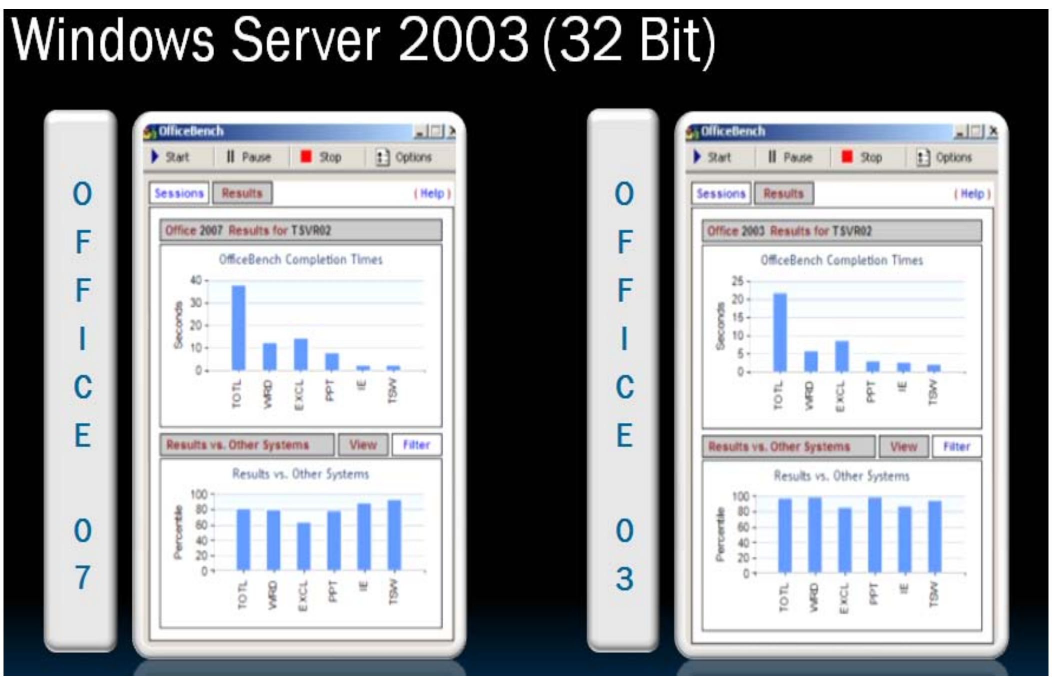

Figure 28. 2004 (32-bit) Results

In general, we noticed that the (64-bit) operating systems performed better in these time-to-completion tests, with Server 2003 (64-bit) taking the top spot. In addition, all operating systems executed the script faster when using Office 2003.

C. PERFORMANCE TEST RESULTS II: PERFORMANCETEST 7.0

Four performance tests were conducted on the server machine, which had been rebooted and allowed to complete its execution of all startup applications and services. The different operating systems tested were:

- Windows Server 2008 (64-bit)
- Windows Server 2008 (32-bit)
- Windows Server 2003 (64-bit)
- Windows Server 2003 (32-bit)

The team expected a repeat of a top score going to Windows Server 2003 (64-bit), however, in this series of tests Windows Server 2008 (64-bit) performed the series of tests well enough to get the highest PassMark score, which was 1995.5. Windows 2003 (64-bit) did very well achieving a score of 1969.2.

Once again, the 64-bit versions of Windows Server outperformed the 32-bit versions in their respective benchmark tests.

Figure 30 shows the overall results from the PerformanceTest benchmarks. The taller the bar in the graph the faster a system executed the PerformanceTest scripted events.

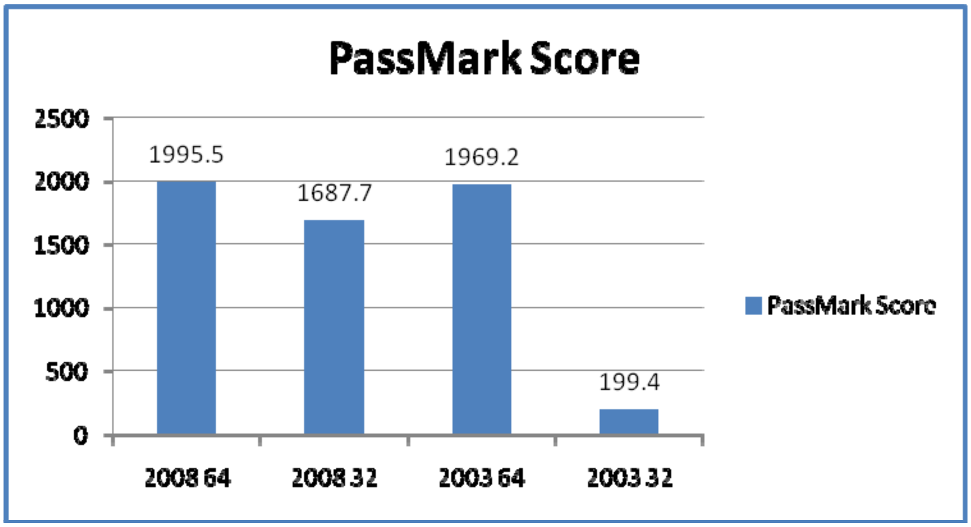

Figure 29. PerformanceTest PassMark Scores

The team feels confident in the results of this test for each of the Windows Server 2008 versions. All scripted events ran as expected in the 2008 variations. However, it is important to note that the scores for the Windows 2003 versions are not correctly calculated due to a problem with 3D acceleration not functioning correctly. In particular, the results for Windows Server 2003 (32-bit) are significantly impacted by the script slowing down to a snail's pace after PerformanceTest was unable to execute its 3D graphics tests. As a result, the score of 199.4 is not representative of the operating systems capabilities.

The results in Table 4 are produced in the same sequence in which the tests were run. It is immediately evident that not all tests that were executed due to the DirectX issues on the 2003 operating systems. However, we feel it is still useful data to compare the two Server 2008 operating systems. From this we can see that Server 2008 (64-bit) received a mark of 1999.5 while the 32-bit version received a lower mark or 1687.7.

For this reason we thought it would be appropriate to include the drilldown numbers that PerformanceTest provides (Table 4). The crossed out results are the individual test that we do not deem credible.

Test Performed	Server 2008 (64-bit)	Server 2003 (64-bit)	Server 2003 (32-bit)	Server 2003 (32-bit)
CPU - Integer Math	2595.6	2649.7	611.4	623.2
CPU - Floating Point Math	3053	3090.9	2503.7	2520.4
CPU - Find Prime Numbers	1244.4	1254.9	1230.5	1230.9
CPU - SSE	28	37.7	18.8	19
CPU - Compression	9000.9	9260.8	8953.8	8936.5
CPU - Encryption	29.1	29.7	27.2	28.4
CPU - Physics	486.5	483.1	455.9	457.2
CPU - String Sorting	4588.2	4488.8	4777.3	4737.4
Graphics 2D - Solid Vectors	7.9	14.2	6	0.3
Graphics 2D - Transparent Vectors	7.4	0.6	5.9	0.3
Graphics 2D - Complex Vectors	161.7	199.3	136.8	186.3
Graphics 2D - Fonts and Text	178.6	332.4	158.4	323.4
Graphics 2D - Windows Interface	149.2	1041.8	138.7	542.6
Graphics 2D - Image Filters	353.5	349.7	545.9	542.2
2D Graphics - Image Rendering	608	609.8	424	
Graphics 3D - Simple	559.7		562	
Graphics 3D - Medium	197		196.6	
Graphics 3D - Complex	34.7		34.7	
Memory - Allocate Small Block	5347.9	~~5678.2~~	3998.8	~~421.9~~
Memory - Read Cached	2819.2	~~2829.7~~	2488.7	~~5159.9~~
Memory - Read Uncached	2680.9	~~2687.1~~	2260.6	~~2496.7~~
Memory - Write	2906.7	~~2889.9~~	2703.2	~~2288.1~~
Memory - Large RAM	8928.5	~~9312.2~~	2305.9	~~2675.2~~
Disk - Sequential Read	65.6	~~93.6~~	75.6	~~2323.8~~
Disk - Sequential Write	69.2	~~1.4~~	76.8	~~3.2~~
Disk - Random Seek + RW	2.7	~~2.5~~	2.7	~~1.4~~
CD - Read	2.7	~~2.8~~	2.8	~~1.4~~
CPU Mark	7716.7	~~8497.6~~	5235.8	~~2.8~~
2D Graphics Mark	886.3	~~1078.6~~	727.7	~~5277.6~~
Memory Mark	3108.8	~~3221.2~~	1298.4	~~505.7~~
Disk Mark	497.1	~~352.4~~	560.8	~~1356.6~~
CD Mark	329.8	~~339.8~~	341.8	~~21.7~~
3D Graphics Mark	462.8		463.3	~~340.5~~
PassMark Rating	1999.5	~~1969.2~~	1687.7	~~199.4~~

Table 4. PerformanceTest Drilldown Results

Because the tests, (for both Server 2003 operating systems) starting at and below "Graphics Test 3D – simple" are not valid due to the scripts problem with 3D graphics tests, we were left with two valid marked areas from which we could still compare across all four operating systems: Disk Tests and 2D Graphics. We weighted these two areas in the same manner as was listed in the benchmark descriptions of this document. We used the relative weights listed in Table 5. These weights are based relative to the original PerformanceTest weights.

Test Suite	**Weighting**
Disk	60%
2D Graphics	40%
Total	**100%**

Table 5. Modified PassMark Rating Weighted Averages (After Robinson, 2008)

Using the following formula to produce mark scores (*weight / sub-score)* and the directions provided in the benchmark description of this document we calculated our own amalgamated score, which is represented in Figure 30. The larger the score, the better the operating system was able to perform the benchmark scripts with regard to speed of completion.

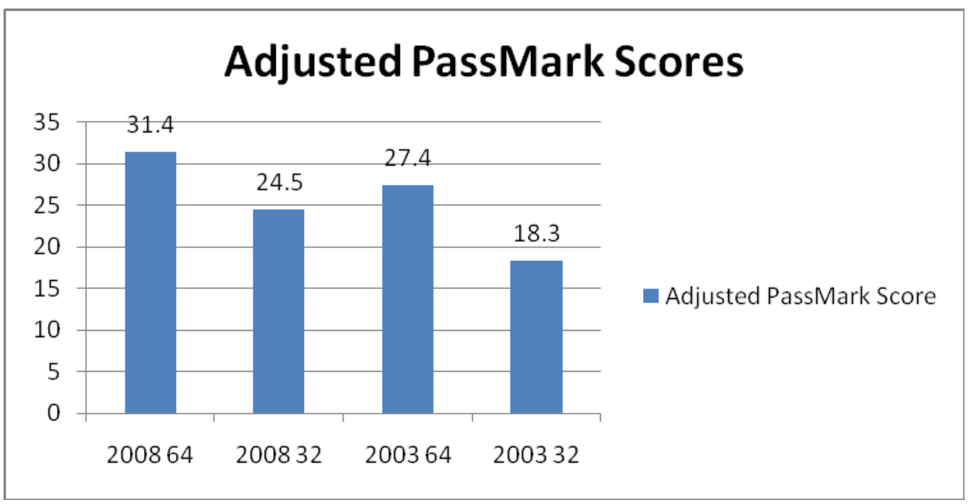

Figure 30. Adjusted PerformanceTest Results

The adjusted results only make use of the CPU and 2D Graphics mark tests, however, it is interesting to note that the results closely mimic the original Passmark scores, with the 2003 (32-bit) version doing much better.

Windows Server 2008 (64-bit) took the top spot with Server 2003 (64-bit) close behind. Once again, the (64-bit) versions did better in the benchmark tests. An updated drilldown table with sub scores is listed in Table 6.

Test Performed	**Server 2008 (64-bit)**	**Weighted Sub-scores**	**Server 2003 (64-bit)**	**Weighted Sub-scores**	**Server 2003 (32-bit)**	**Weighted Sub-scores**	**Server 2003 (32-bit)**	**Weighted Sub-scores**
CPU - Integer Math	2595.6	0.00023	2649.7	0.00023	611.4	0.00098	623.2	0.00096
CPU - Floating Point Math	3053	0.00020	3090.9	0.00019	2503.7	0.00024	2520.4	0.00024
CPU - Find Prime Numbers	1244.4	0.00048	1254.9	0.00048	1230.5	0.00049	1230.9	0.00049
CPU - SSE	28	0.02143	37.7	0.01592	18.8	0.03191	19	0.03158
CPU - Compression	9000.9	0.00007	9260.8	0.00006	8953.8	0.00007	8936.5	0.00007
CPU - Encryption	29.1	0.02062	29.7	0.02020	27.2	0.02206	28.4	0.02113
CPU - Physics	486.5	0.00123	483.1	0.00124	455.9	0.00132	457.2	0.00131
CPU - String Sorting	4588.2	0.00013	4488.8	0.00013	4777.3	0.00013	4737.4	0.00013
Graphics 2D - Solid Vectors	7.9	0.05063	14.2	0.02817	6	0.06667	0.3	1.33333
Graphics 2D - Transparent Vectors	7.4	0.05405	0.6	0.66667	5.9	0.06780	0.3	1.33333
Graphics 2D - Complex Vectors	161.7	0.00247	199.3	0.00201	136.8	0.00292	186.3	0.00215
Graphics 2D - Fonts and Text	178.6	0.00224	332.4	0.00120	158.4	0.00253	323.4	0.00124
Graphics 2D - Windows Interface	149.2	0.00268	1041.8	0.00038	138.7	0.00288	542.6	0.00074
Graphics 2D - Image Filters	353.5	0.00113	349.7	0.00114	545.9	0.00073	542.2	0.00074
CPU Mark (1/weighted sub score)	22.53	0.04439	26.00	0.03846	17.49	0.05719	17.89	0.05590
2D Graphics Mark (1/weighted sub score)	8.83	0.11321	1.43	0.69957	6.97	0.14353	0.37	2.67153
Adjusted Accumulated Totals	**31.4**		**27.4**		**24.5**		**18.3**	

Table 6. Adjusted PassMark Score Drilldown Table

D. PERFORMANCE TEST RESULTS III: NOVABENCH 3.0.1

While the results garnered from the PerformanceTest tool are helpful in helping us determine which operating system our team would recommend to the NPS GSBPP; our team searched for a similar tool to Performance Test 7.0. This search led us to the

NovaBench tool. The two benchmark applications are similar in the areas they test and both are able to test across operating systems. Although, the level of transparency in how NovaBench produces its amalgamated final score is not at the same level as PerformanceTest.

However, NovaBench had a key option that made it suitable for our testing needs; it allowed us to skip the 3D graphics portion of their test script.

As with the PerformanceTest benchmarks, four benchmarks were conducted on the server machine. Each was rebooted and allowed to complete all of the execution of their respective startup applications and services. The operating systems tested with NovaBench were:

- Windows Server 2008 (64-bit)
- Windows Server 2008 (32-bit)
- Windows Server 2003 (64-bit)
- Windows Server 2003 (32-bit)

The team was interested to see if the trend of (64-bit) dominance would continue and whether or not one of the 2003 operating systems would finally come out on top in a time-to-completion benchmark test.

However, it would not be in this series of tests; the (64-bit) operating system's ability to perform this type of time-to-completion test proved to be better once again.

Figure 31 shows the overall results from the NovaBench benchmarks. The taller the bar in the graph the faster a system executed the NovaBench scripted events.

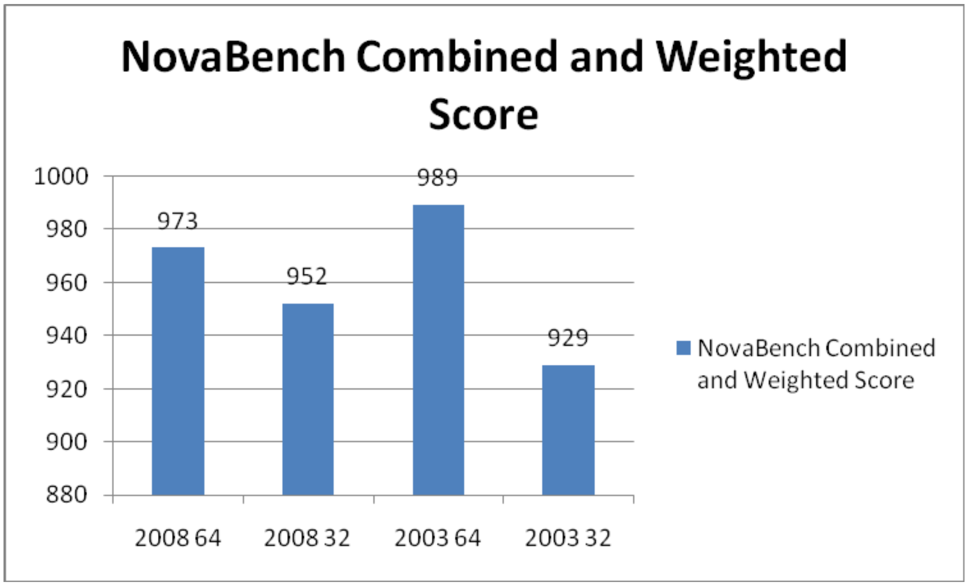

Figure 31. NovaBench Overall Results

The NovaBench scores are derived from a series of individual tests. The individual areas tested are:

- RAM Speed
- CPU Tests
- Floating Point Operations/Second
- Integer Operations/Second
- MD5 Hashes Generated/Second
- Graphics Tests [2]
- 3D Frames Per Second
- Hardware Tests
- Primary Partition Capacity
- Drive Write Speed

NovaBench results for Windows Server 2008 (64-bit) version are:

- 12279MB System RAM (Score: 209)
- RAM Speed: 9820 MB/s

[2] This test was not performed by deselecting the Graphics Test option. This was a key option for this test due to the difficulties we experienced with 3D graphics tests on Windows Server 2003 operating systems.

- CPU Tests (Score: 711)
- Floating Point Operations/Second: 2055828723
- Integer Operations/Second: 808416712
- MD5 Hashes Generated/Second: 10429204
- Hardware Tests (Score: 53)
- Drive Write Speed: 192 MB/s

While this format and data are useful, for the purpose of comparing individual operating system against one another, our team produced bar charts of each area (Figure 32). This approach serves to give an easy to understand and quick view of the results across the four server operating systems.

1. NovaBench Drilldown Graph: RAM Speed Test

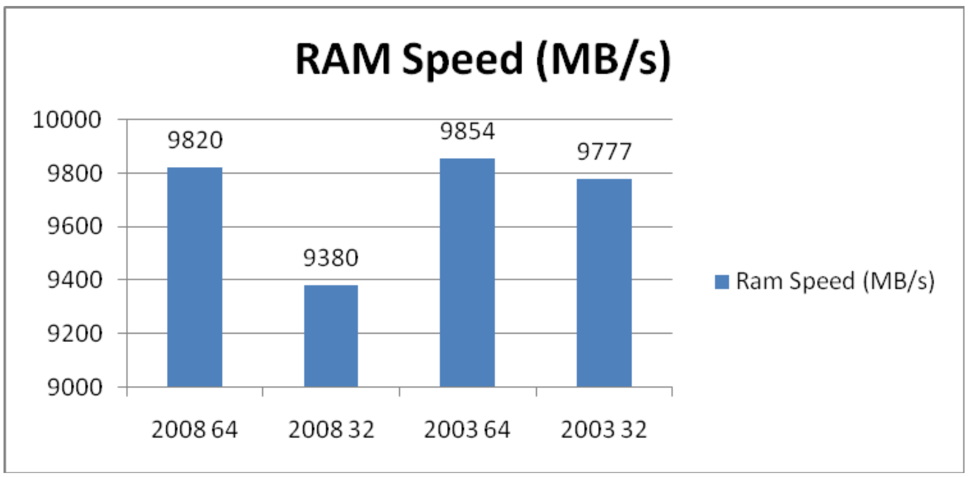

Figure 32. NovaBench RAM Speed Results

It is interesting to see that both 2003 operating systems did very well in this test, with 2003 (64-bit) version performing the best in this test. However, Server 2008 (32-bit) was the overall leader in the NovaBench tests, and scored the lowest by a sizeable margin.

[3] An operation for representing numbers that is too large to be shown as integers.

[4] MD5 Hashes Generated is a method of verifying data integrity.

2. NovaBench Drilldown Graph: CPU Tests

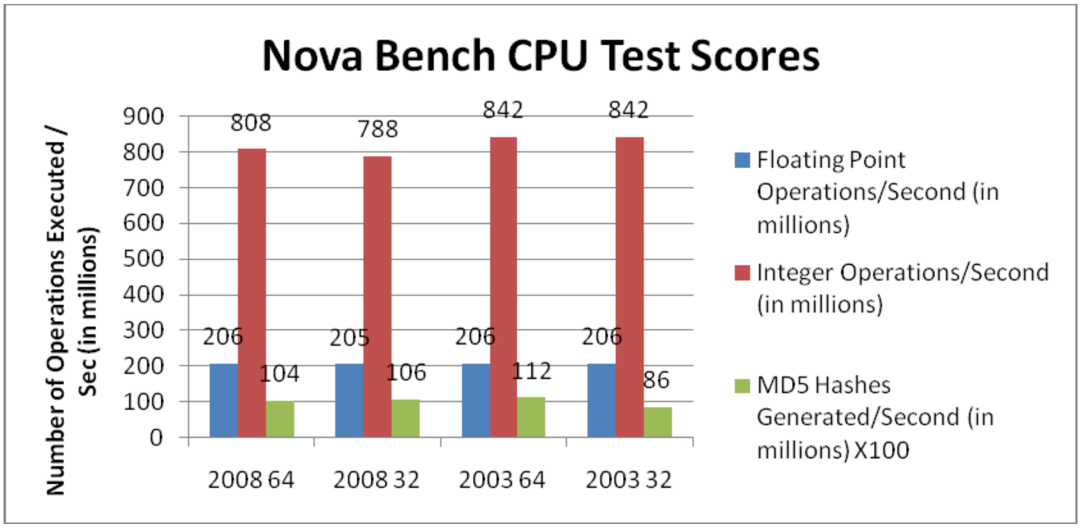

Figure 33. NovaBench CPU Tests Results

Once again, Server 2003 (64-bit) version performed the best in this test, by executing or generating more operations / MD5 hashes during the NovaBench scripted test. The results in Figure 33 are in the millions, so a score of 206 is 206,000,000 operations executed in one second. It is important to note that the numbers of MD5 hashes generated were multiplied by 100 times for the purposes of giving them a viewable scale in Figure 33. A result of 104 in the MD5 hash column is equal to 1,040,000.

3. NovaBench Drilldown Graph: Hardware Test

In this test, Server 2003 (64-bit) version performed the worst (131MB/s). Initially Server 2003 64 tested so badly (23 MB/s) that we rebooted and ran the test a few more times. We finally received consistent scores in the 131 MB/s range that we felt comfortable reporting the result. See Figure 34.

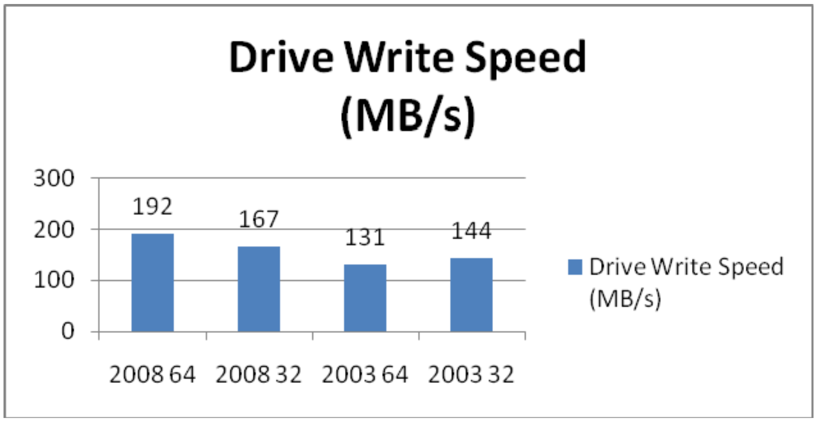

Figure 34. NovaBench Hardware Test Result

4. NovaBench Combined and Weighted Scores

NovaBench also calculates their totaled scores by providing weighted scores to the sub-scores and then sums them up to get a total NovaBench score. From this we can see that Server 2003 (64-bit) version did well enough in the CPU Test areas that it took the top spot in the NovaBench benchmark tests overall. It is worth noting once again that not all went well with Server 2003 (64-bit) version, its poor ranking in hardware tests made this a mixed bag of results.

A bar graph summarizing the NovaBench scores is presented in Figure 35. The NovaBench blue bar is the sum of the other three individual tests.

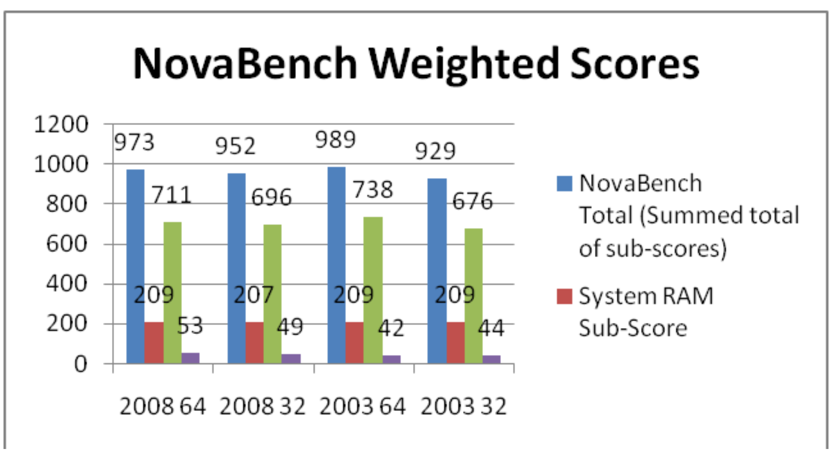

Figure 35. NovaBench Combined and Weighted Scores

Continuing with the theme established in previous tests, the 64-bit operating systems once again outperformed the 32-bit versions in this time-to-completion benchmark.

E. PERFORMANCE TEST RESULTS IV: STARTUP/SHUTDOWN TIMES

As noted previously, we fully admit that there are unique differences in the startup applications and services that impact these results. The team used as close to an out of the box installation as we could with the exception of installing the tools we needed for this project and the services needed to establish our ad hoc server, and applying all available patches from Microsoft prior to testing.

Because of the need to partition the hard drive in to four sections, we began the stopwatch at the time we hit enter from the boot manager and stopped it when we reached the control-alt-delete part of the startup process. We then continued the stopwatch again after we entered our credentials and hit enter at the login screen. This was done to discount any human input in to the test. We stopped the clock at the moment we saw the GUI appear. The two times were then combined to reach a total startup time.

For the shutdown time it was a simple matter of beginning the count from the moment the shutdown option was selected and stopping the clock when the power light went out.

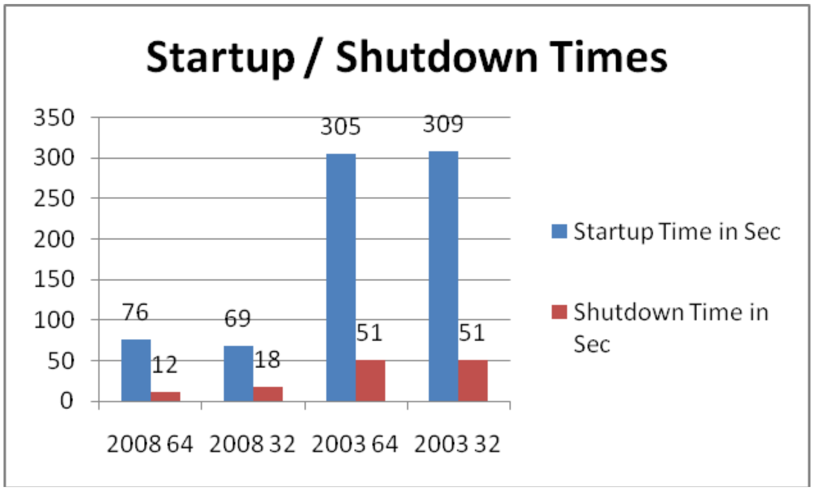

Figure 36. Startup and Shutdown Results (Measured in Seconds)

The results in Figure 36 appear to be influenced by the operating system and not the amount of bits. This is the first test with these types of results and we can see that Server 2008 operating systems easily performed better in time-to-completion in this test.

F. PERFORMANCE TEST RESULTS V: PERFORMANCE MONITOR

For this test, we used Windows Reliability and Performance Monitor (PERFMON) to test resource usage of server operating systems running a commonly used statistics program (Stata 11 I/C). This test was run from a client machine using remote desktop connection to the server. All servers used Terminal Services with DNS. The client machine used two different Operating Systems; the combinations of test setups are:

- Windows Server 2008 (64-bit) with Windows XP Professional SP3
- Windows Server 2008 (64-bit) with Windows 7 Professional
- Windows Server 2008 (32-bit) with Windows XP Professional SP3
- Windows Server 2008 (32-bit) with Windows 7 Professional
- Windows Server 2003 (64-bit) with Windows XP Professional SP3
- Windows Server 2003 (64-bit) with Windows 7 Professional
- Windows Server 2003 (32-bit) with Windows XP Professional SP3
- Windows Server 2003 (32-bit) with Windows 7 Professional

Performance Monitor monitored the following three resources:

- Percentage Processor Time
- Pages/sec
- Average Disk Queue Length.

Each test was run with the respective server having Stata installed on its hard drive; while each respective client sent input and requested resources from the server. In other words, all the keyboard strokes and mouse movements were done on the client to control the actions of Stata, while the server ran the application and provided resources.

There are differences between the versions of Windows Reliability and Performance Monitor in Server 2008 and 2003. The difference we would like to note is that in 2008 there is a function to pause and start the monitor. However, in 2003 no

function like this exists. It is because of this small difference that all PERFMON graphs dealing with Server 2003 have a start and stop line delineated to show the exact time we started Stata and when we concluded.

The results are provided in the line graph format provide by PERFMON. In general the shorter and less frequent the spikes are the better the efficiency in each area. We conceded that the results of these graphs are subjective; however, it is our intent to give an educated opinion of which combinations did better overall in this kernel performance test. We list results in the order of rank from best use of resources to worst.

1. Server 2008 (32-bit) Version Along With a Windows 7 Client

The first consideration used was the top performing combination. Extremely efficient scores during the startup of Stata caused us to redo this test, but we ended up with similar results. Processor activity (red line) and system delays (blue line) are minimal and the average number of read and write requests queued in this sample (green line) are on par or below the other combinations tested.

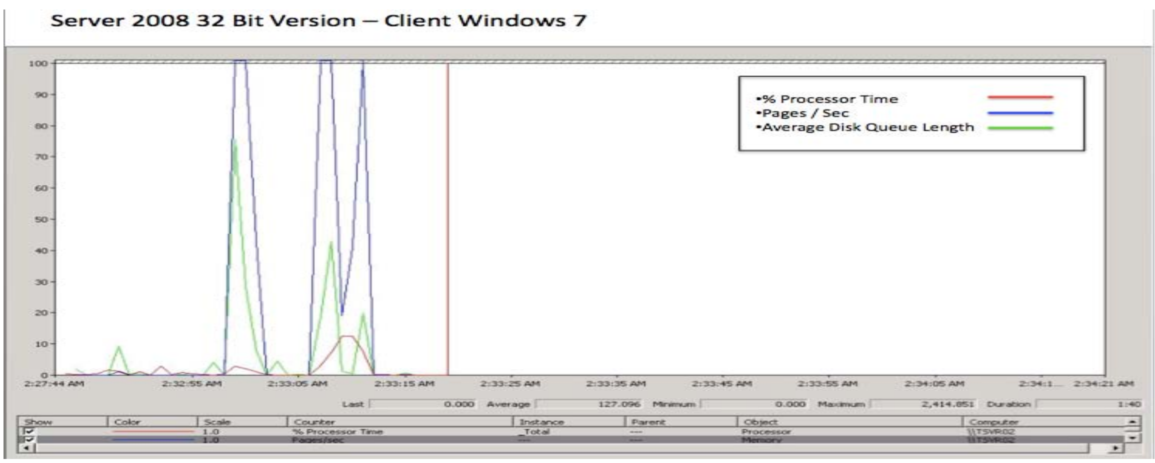

Figure 37. PERFMON Results for Server 2008 (32-bit) Version and Windows 7

2. Server 2003 (64-bit) Version Along With a Windows XP Client

The difference between numbers two and three in this list came down to how well Server 2003 (64-bit) along with Windows XP did in minimizing system delays (represented in blue line).

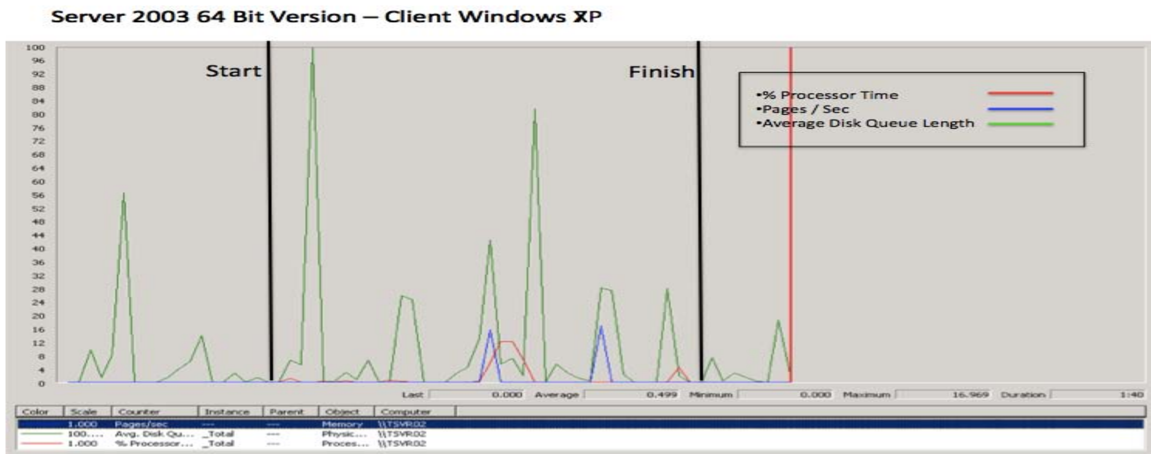

Figure 38. PERFMON Results for Server 2003 (64-bit) Version and Windows XP

3. Server 2008 (64-bit) Version Along With a Windows 7 Client

Here you can see the higher representation of system delays (blue line) mentioned in the previous paragraph. Although there are more instances of system delays in the startup of Stata and when opening the test file, we gave this combination the nod over #4 on this list due to its performance during chart generation portion of the test.

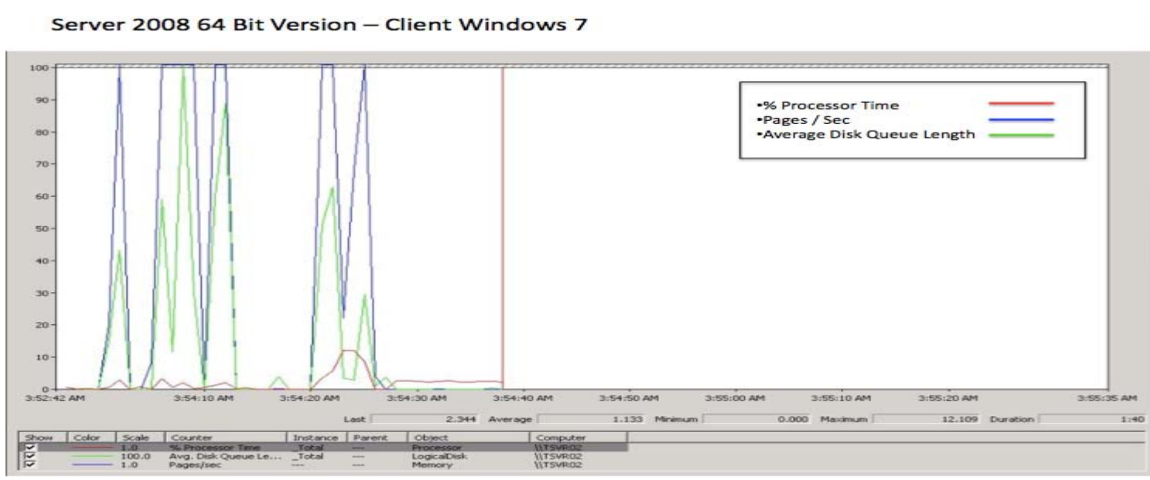

Figure 39. PERFMON Results for Server 2008 (64-bit) Version and Windows 7

4. Server 2003 (64-bit) Version Along With a Windows 7 Client

The following two rankings are also extremely close; with the determining variable being a lower spike in system delays during the time Stata was asked to open the test file (see first blue spike, Figure 40).

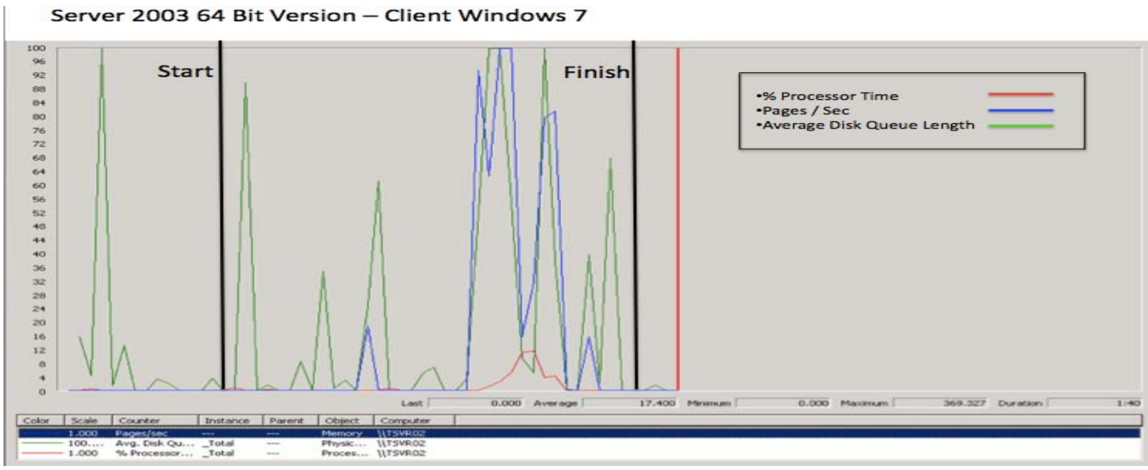

Figure 40. PERFMON Results for Server 2003 (64-bit) Version and Windows 7

5. Server 2003 (32-bit) Version Along With a Windows 7 Client

Here you can see how close these results were with #4. Please note the higher initial system delay spike (represented in blue line) when Stata was opening the test file. There is slightly higher processor activity (red line) and that point as well.

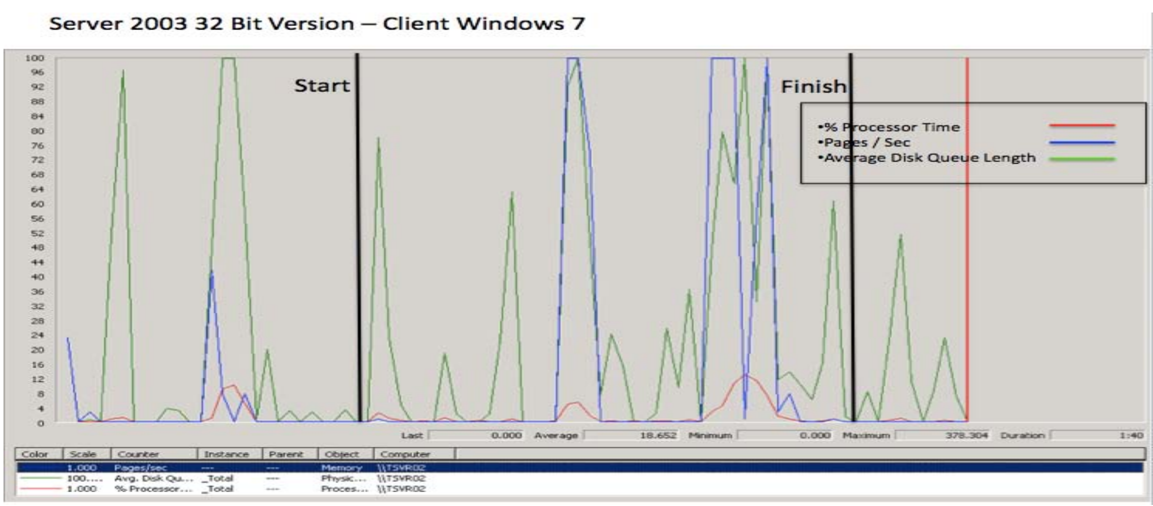

Figure 41. PERFMON Results for Server 2003 (32-bit) Version and Windows 7

6. Server 2003 (64-bit) Version Along With a Windows XP Client

Higher occurrences of processor activity and system delays in this combination led to a ranking of sixth place.

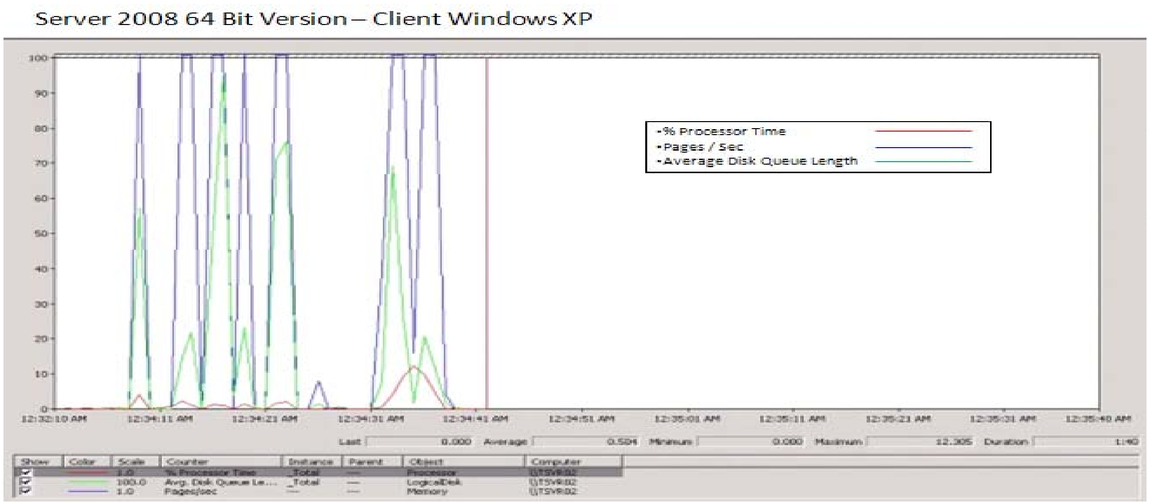

Figure 42. PERFMON Results for Server 2008 (64-bit) Version and XP

7. Server 2008 (32-bit) Version Along With a Windows XP Client

Very similar results to our sixth ranked combination, however, we were unable to generate results for Average Disk Queue Length. That flaw in testing knocked this combination down in the list of results.

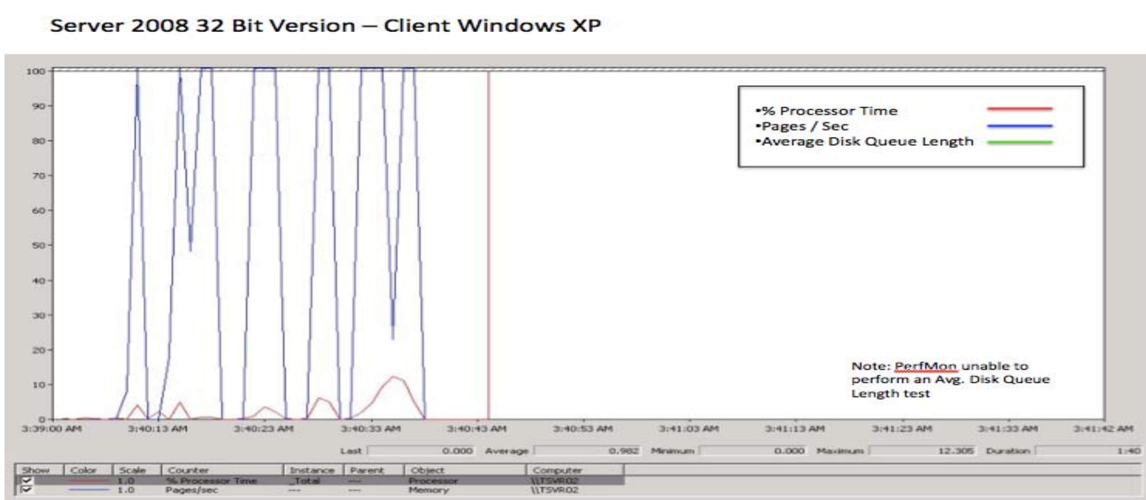

Figure 43. PERFMON Results for Server 2008 (32-bit) Version and Windows XP

8. Server 2003 (32-bit) Version Along With a Windows XP Client

The highest amount of processor activity, system delays and average read and write requests seen in any of our tests.

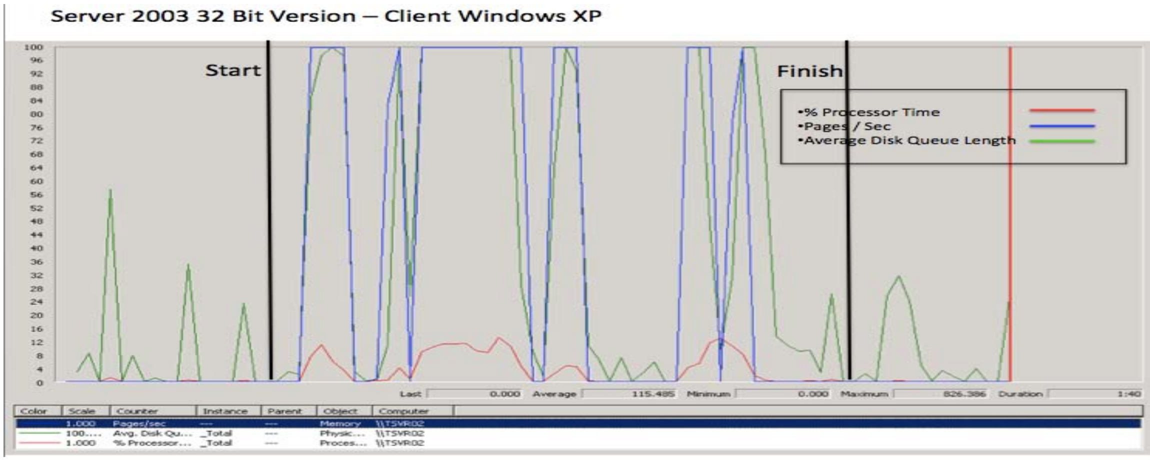

Figure 44. PERFMON Results for Server 2003 (32-bit) Version and XP

9. Consolidated List of PERFMON Results

The following list summarizes the rankings of how well each combination performed in this benchmark.

- Windows Server 2008 (32-bit) - Windows 7 Client
- Windows Server 2003 (64-bit) - Windows XP Client
- Windows Server 2008 (64-bit) - Windows 7 Client
- Windows Server 2003 (64-bit) - Windows 7 Client
- Windows Server 2003 (32-bit) - Windows 7 Client
- Windows Server 2003 (64-bit) - Windows XP Client
- Windows Server 2008 (32-bit) - Windows XP Client
- Windows Server 2003 (32-bit) - Windows XP Client

From here, we see some general patterns:

- The 64-bit operating systems generally placed higher
- Windows 7 Clients generally placed higher

It is interesting to note however that the top spot was awarded to a 32-bit server operating system. This is the only occurrence of a 32-bit operating system placing in the top spot in any of our tests.

G. PERFORMANCE TEST RESULTS VI: LIST COMMAND–TIME TO COMPLETION

For our final test, we performed a time-to-completion test with STATA. We ran the LIST command on a 27Mb file. The list command performs a function similar to a PRINT command where each list is represented on the screen for the user to review. We found that this basic command (using a large file) took an adequate amount of time for measuring time-to-completion of a task using a common stopwatch.

The following setups were tested with this benchmark.

- Windows Server 2003 64-bit with Windows XP
- Windows Server 2008 64-bit with Windows XP
- Windows XP on it own
- The results are as follows:
- Server 2003 64-bit (client = XP): 26 min 42 sec
- Server 2008 64-bit (client = XP): 26 min 17 sec
- Client (Windows XP) on its own (no application server): 39 min 02 sec

This test demonstrates how a server application impacts / mitigates time intensive operations with an application that requires a heavy amount of RAM resources.

VI. FINDINGS AND RECOMMENDATIONS

A. CONSOLIDATED RESULTS

In order to provide a final recommendation we found it useful to provide perspective on how a given operation system performed over all. We chose to do this by providing a table of amalgamated results. We first ranked how each system did in each individual benchmark test in Table 7.

	OfficeBench MS Office 7	Performance Test (adjusted results)	NovaBench	Startup	Shutdown	PERFMON w/ Win 7	PERFMON w/ Win XP
2008 32	3	3	3	1	2	1	3
2008 64	2	1	2	2	1	2	1
2003 32	4	4	4	4	4	4	4
2003 64	1	2	1	3	3	3	2

Table 7. Overall Server OS Ranking

While this format is interesting, it provides little context. Not each test can be considered equal. Based on the needs of the GSBPP to provide a server that minimized time-to-completion of task performed on commonly used business applications and how well the benchmark tools performed on our server and client architecture we assigned the following relevancy weights.[5]

	Weighted %
Office Bench	25%
PerformanceTest (adjusted results)	15%
Nova Bench	20%
Startup	5%
Shutdown	5%
PERFMON w/ Win 7	15%
PERFMON w/ Win XP	15%
Total	100%

Table 8. Benchmark Application Weights

[5] Only Microsoft Office 2008 results are used to calculate our final results. At the time of this writing the NPS GSBPP used Office 2008 and had no intention of reverting to Office 2003. As a result we did not include Office 2003 results in any of our final calculations or figures.

In order to provide a quick and discernable measurement we applied the following formula:

$$(1/\ (\text{weighted }\%\ /\ \text{rank}))\ X\ 100$$

We divide the weighted percentage from Table 7 by the respective ranking from Table 8. This produces a number that gets smaller the better an operating system did, so we invert the number by dividing one by the product of *(weighted % / rank).* Finally, in order to give the result some scale we multiplied the product of *1/ (weighted % / rank)* by 100. This provides results that increase with how well an operating system did by rank in each of our benchmark tests. Finally, we total each of the sub scores to receive overall weighted results. These results are represented in Table 9.

	Office Bench MS (Office 7) Sub Score	Performance Test (adjusted results) Sub Score	Nova Bench Sub Score	Start up Sub Score	Shut down Sub Score	PERFMON w/ Win 7 Sub Score	PERFMON w/ Win XP Sub Score	Total Weighted Score
2008 (32-bit)	8.33	5.00	6.67	5.00	2.50	15.00	5.00	47.50
2008 (64-bit)	12.50	15.00	10.00	2.50	5.00	7.50	15.00	67.50
2003 (32-bit)	6.25	3.75	5.00	1.25	1.25	3.75	3.75	25.00
2003 (64-bit)	25.00	7.50	20.00	1.67	1.67	5.00	7.50	68.33

Table 9. Overall Server Operating System Weighted Total Scores

Both 64-bit versions of Windows Server operating system outperformed the 32-bit versions in the time-to-completion performance tests and benchmarks we conducted. However, it is interesting to note that Windows Server 2003 (64-bit) did slightly better overall in our weighted results than Windows Server 2008 (64-bit) version.

We feel it necessary to stress that our performance tests were done with easily accessible tools that are not necessarily SPEC industry standards. Also, our results are largely based on time-to-completion analysis and do not account for other pertinent areas of measure. Finally, our limited reliability benchmarks with Microsoft's Reliability and Performance Monitor application were subjective and could be interpreted differently by an expert.

However, the benchmark tests reported here accomplished the goal we set for ourselves, which was to provide the GSBPP with solid analysis on which Server Operating System performs better in time-to-completion benchmark tests. From the results reported we feel confident in providing a recommendation.

B. RECOMMENDATION

Our team confidently recommends to the Naval Postgraduate School, Graduate School of Business and Public Policy, Windows Server 2008 (64-bit) operating system.

With regard to how the client operating systems did Windows 7 did better in the limited tests we performed, however Windows XP was not far behind. Our team recommends Windows 7 client operating system.

1. Team Insight

Despite Server 2003 (64-bit) version having a higher score in our weighted final results, it is the experience of this team from working with both of these operating systems during this project that Server 2008 is a more polished, evolved and refined product. We felt that results were close enough between Server 2003 (64-bit) and Server 2008 (64-bit) that the intuitive functions and refinements in Server 2008 (64-bit) were enough to garner its selection as our recommendation.

Our team experienced longer set up times and challenges with Server 2003 products than with Server 2008. In particular, establishing the ad hoc client–server test network was particularly challenging. In contrast setting up a network with Server 2008 was simple and intuitive.

Admittedly, our team is not made up of experienced Network Administrators, we are Business School students. But, our team was able to use Server 2008 products with little to no issues, while additional challenges with the Graphics Processing Unit and DirectX 3D acceleration Server 2003 impacted project analysis and server performance.

Either operating system Windows Server 2003 (64-bit) version or Server 2008 (64-bit) version will serve the GSBPP well. However, consideration should be taken in to account that Server 2003 is a previous generation technology being replaced by Server

2008. One should contemplate if equal support will be given to Server 2003 in the future as Microsoft moves forward with Server 2008.

For these reasons, analysis and experience we recommend the GSBPP select Windows Server 2008 (64-bit) version as the operating system for its application server architecture.

LIST OF REFERENCES

Allen, M. (2001). *Windows 2000 Terminal Services*. Retrieved on 08/08/2010, from <http://www.comptechdoc.org/os/windows/win2k/win2kterminal.html>

Alliegro, C. (2003). *Performance Improvements in Windows Server*. Retrieved on 08/05/2010, from Directions on Microsoft <www.directionsonmicrosoft.com>

Fortin, S. (2001). *Windows XP Performance*. Retrieved on 08/10/2010, from <http://technet.microsoft.com/en-us/library/bb457057.aspx>

Goktepe, M. (2002). *Windows XP Operating System Security Analysis*. Thesis, Monterey, CA: Naval Postgraduate School.

Graduate School of Business and Public Policy (Naval Postgraduate School) (2010). *About: The Graduate School of Business & Public Policy*. Retrieved on 08/13/2010, from <http://www.nps.edu/Academics/Schools/GSBPP/About/About.html>

Gustafson, J. (2004). "Purpose Based Benchmarks." *International Journal of High Performance Computing Applications*.

Hockney, R. (1996). *The Science of Computer Benchmarking.* Philadelphia, PA: Society for Applied Mathematics.

Kennedy, R. C. (2009a). *Clocking Windows netbook performance*. Retrieved on 08/14/2010, from http://podcasts.infoworld.com/d/hardware/clocking-windows-netbook-performance-883?_kip_ipx=1177119066-1281460794

Kennedy, R. C. (2009b). *OfficeBench 7: A cool new way to evaluate PC performance*. Retrieved 08/14/2010, from http://www.networkworld.com/news/2009/100709-officebench-7-a-cool-new.html

LaPierre, N. (2010). *About NovaBench*. Retrieved 08/08/2010, from http://novabench.com/about.php

Madden, B. S. (2004). *Terminal Services for Microsoft Windows Server 2003.* Washinton D.C.: BrianMadden.com.

Microsoft Corporation. (2010a). *Computers that are running Windows XP Service Pack 2 and that are equipped with multiple processors that support processor power management features may experience decreased performance*. Retrieved 08/10/2010, from Microsoft: http://support.microsoft.com/kb/896256

Microsoft Corporation. (2008a). *How Windows Server 2008 Delivers Business Value.* Microsoft Corporation.

Microsoft Corporation. (2001). *Microsoft Announces Windows XP and Office XP*. Retrieved 08/10/2010, from Microsoft: http://www.microsoft.com/presspass/press/2001/feb01/02-05namingpr.mspx

Microsoft Corporation. (2007a). Microsoft Windows Server 2008 Licensing Guide. Microsoft.

Microsoft Corporation. (2007b). *System requirements for Windows XP operating systems*. Retrieved 08/17/2010, from Microsoft: http://support.microsoft.com/kb/314865

Microsoft Corporation. (2008b). *Windows 7: A First Look for IT Pros*. Retrieved 08/10/2010, from Microsoft: http://technet.microsoft.com/en-us/library/dd266801.aspx

Microsoft Corporation. (2006). *Windows Server 2003 Terminal Server licensing issues and requirements for deployment*. Retrieved 08/21/2010, from http://support.microsoft.com/kb/823313

Microsoft Corporation. (June 2008c). *Windows Server 2008 Power Savings.* Microsoft Corporation.

Microsoft Corporation. (2010b). *Windows Server 2008 R2 System Requirements*. Retrieved 08/17/2010, from Microsoft: http://www.microsoft.com/windowsserver2008/en/us/system-requirements.aspx

Microsoft Corporation. (2010c). *Windows Vista system requirements*. Retrieved 08/18/2010, from Microsoft: http://windows.microsoft.com/en-us/windows-vista/products/system-requirements

Microsoft, Corporation. (2008d). Reliability and Performance Monitor Help File.

Montehermosa, R. C. (2004). *Microsoft Windows Server 2003: Security Enhancements and New Features.* Thesis, Monterey, CA: Naval Postgraduate School.

Pfeiffer Consulting. (2007). *User Interface Friction Benchmarks: Windows Vista vs. Windows XP and Mac OS X.* Pfeiffer Consulting.

Reese, G. (2000). *Database Programming with JDBC and Java, Second Edition.* Sebastopol, CA: O'Reilly Media.

Robinson, I. (2008). *Benchmarking Passmark PerformanceTest*. Retrieved 08/07/2010, from http://www.techspot.com/downloads/1110-performancetest.html

Robinson, I. (2010). PerformanceTest Help Booklet. Sydney, Australia.

Romero, D. (2008). *NovaBench*. Retrieved 08/08/2010, from http://novabench.software.informer.com

Salamon, A. (2008). *DNS Resources Directory*. Retrieved 08/08/2010, from http://www.dns.net/dnsrd

Sangani, K. (2009a). 7 Cardinal Virtues. *Engineering & Technology*, 57.

Sangani, K. (2009b). Lucky Seven? *Engineerign & Technology*, 32–33.

Smith, A. (2009). *Benchmarks: Windows 7 RTM versus Vista, XP*. Retrieved 08/13/2010, from ZDNet: http://www.zdnet.com/blog/btl/benchmarks-windows-7-rtm-versus-vista-xp/22006?tag=mantle_skin;content

Stanek, W. (2008). *Windows Server 2008 Inside Out.* Redmond, Washington: Microsoft Press.

Tayntor, C. B. (2006). *Successful Packaged Software Implementation.* Boca Raton, FL: Auerbach Publications.

The Tolly Group. (2008). *Enhanced Network Performance with Microsoft Windows Vista and Windows Server 2008.* The Tolly Group, Inc.

Truttman, J. (2009). *OfficeBench 7: A cool new way to evaluate PC performance*. Retrieved 03/13/2010, from http://www.cio.de/news/cio_worldnews/2210334/index2.html

Webmaster, S. (2010). *StataCorp About Page*. Retrieved 08/07/2010, from http://www.stata.com/about

Wren, D. (2005). *Company Profile for PassMark Software Pty Ltd, T/A Wrensoft*. Retrieved 08/08/2010, from http://findarticles.com/p/articles/mi_m0EIN/is_2005_March_11/ai_n12418665/

Wren, D. (2010). *PerformanceTest, Easy Benchmark Testing*. Retrieved 08/08/2010, from http://www.passmark.com/products/pt.htm

THIS PAGE INTENTIONALLY LEFT BLANK

INITIAL DISTRIBUTION LIST

1. Defense Technical Information Center
   Ft. Belvoir, Virginia

2. Dudley Knox Library
   Naval Postgraduate School
   Monterey, California

3. Dr. Douglas Brinkley
   Naval Postgraduate School
   Monterey, California

4. Professor Albert Barreto
   Naval Postgraduate School
   Monterey, California